

5.6 Desktop Classes

5.6.1 Desktop Classes Overview

The Desktop CSCI provides a desktop environment, which consists of a set of user and client files (called desktop objects, each with a representative icon), a pop-up menu for quick access to specific client applications, specific operations defined for desktop

objects, including drag and drop, double-click to open, copy, delete, and rename, and a desktop manager, which manages the set of desktop objects. Desktop objects include, but are not limited to: search parameters, result-sets, documents, subscriptions, request-sets,

data granules, project folders, application programs, and configuration files. Managing the desktop objects includes viewing them in a hierarchical format, viewing their attributes, e.g., name, creation date, and status. The status is applicable only to certain classes of objects, such as product or search requests, and provides a convenient means of providing information about asynchronous requests from the user.

In Release A, the desktop manager will have two versions. There will be a custom developed desktop manager built with X-Windows Version 11 (X11), Motif, and the Common Desktop Environment (CDE) toolkits. This custom desktop is for users who do not use CDE. At Release A, this is most likely a large part of the community. The other version of the desktop manager will be an off-the-shelf (OTS) version using CDE in its native environment. Thus, the desktop objects will reside on the CDE desktop manager. This will be accomplished by making the files representing the desktop objects comply with the CDE standard. Thus the implementation of the desktop object classes will not change no matter which desktop manager is being used by the user. In Release B, it is likely that a much higher number of users will be using the OTS CDE solution. By later releases, it is planned to eliminate the custom desktop manager based on the number of users that have migrated to the OTS solution.

The desktop is either represented in hierarchical format or iconic format. The hierarchical format is similar to the Microsoft Windows implementation of the File Manager or the Macintosh Desktop

implementation when the user selects View By Name. The iconic format is similar to the Microsoft Windows Program Manager implementation or the Macintosh Desktop when in View by Icon. The desktop manager contains desktop objects. There are three types of desktop objects; Container objects, Application objects and Document objects.

Each desktop object will have zero or more actions attached to it. The default action for an object will be executed by double clicking on the icon that represents the object. Other actions can be performed from the Actions pulldown menu. For example, an icon representing a universal reference (UR) to a granule can be ordered by double-clicking the icon. The default action for the granule UR icon is to start the PRT with the associated UR. Another action that could be performed on a granule UR icon include starting the ESST with the associated granule, which would show the granule on the results screen. This secondary action would be performed by selecting the icon and then pulling down the Actions menu and selecting ESST. Each desktop object is represented with a specific icon on the desktop. CDE provides configuration files that associate appropriate attributes with the object types. Thus, the set of actions can be changed by modifying the CDE configuration files.

Users can add their own tools to the desktop by modifying the appropriate CDE configuration files. These files would have to be modified to create an object type and specify the appropriate actions as well as Drag and Drop specifications. The X/Open CDE specification provides details on how to change these configurations.

5.6.2 Desktop Class Descriptions

5.6.2.1 Class CIDtActions

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: CIDtDesktop_Object(Public) HAS - A Desktop Object has default and other actions that can be performed on the object.

Description:

An Action is an Event to be performed by the Desktop on user's or system's request. For example; OpenApplication, CloseApplication, DragIcon, CopyObject, OpenNewWindow and RefreshScreen etc. This Class contains implementations of all actions supported by Desktop objects. The set of Actions to be supported by a Desktop object will be based on its type.

Attributes:

Action_Name

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Name of the action

Operations:

```
int ImplementActions
(Action_Name:char*,ClDtDesktop_Object.NameOfObj:char*)
    Privilege: Public
    No Inheritance
    Implementation of the given action
```

5.6.2.2 Class ClDtApplicationObject**Synopsis:**

Parent Class: ClDtDesktop_Object
Is Not A Distributed Object
Is Associated With:
This class is derived from the class ClDtDesktop_Object

Description:

Application object is a type of desktop object. It provides default behaviour for objects which represent executable programs.

Attributes:

ApplName

Privilege: Private
Data Type: char*
Default Value: NOT IDENTIFIED
No Inheritance
Name of Application Object

ApplObjType

Privilege: Private
Data Type: int
Default Value: NOT IDENTIFIED
No Inheritance
Type of Application Object

IconForApplObj

Privilege: Private
Data Type: char*
Default Value: NOT IDENTIFIED
No Inheritance
Filename of the pixmap to be used as icon for Application object

IconForDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Filename of the pixmap to be used as icon for Desktop Object

NameOfDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Desktop Object Name

Operations:

int AcceptDrop (SrcName:char*,ApplName:char*)

Privilege: Public

No Inheritance

Accepts the related data files as drop input and opens the application.

void* GetAction

(Action_Name:char*,ApplName:char*,ApplObjType:int)

Privilege: Public

No Inheritance

Get the function which implements the specified action for the application object

int OpenAppObj (ApplName:char*)

Privilege: Public

No Inheritance

Open the Application

int CloseDtObj ()

Privilege: Public

Inherited From: CIDtDesktop_Object

Close the Desktop object

int CopyDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

Copy the Source Desktop Object to the specified target

int DragDtObj (DtObjType:int)

Privilege: Public

Inherited From: CIDtDesktop_Object

void* GetAction

(Action_Name:char*,NameOfDtObj:char*,DtObjType:int)

Privilege: Public

Inherited From: CIDtDesktop_Object

Get an action(s) from CIDtActions objects which is related to the desktop object of given type.

int MoveDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

Move the Source Desktop Object to the specified target

int OpenDtObj (DtObjType:int,IconForDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

5.6.2.3 Class CIDtContainerObject

Synopsis:

Parent Class: CIDtDesktop_Object

Is Not A Distributed Object

Is Associated With:

This class is derived from the class CIDtDesktop_Object

Description:

A type of Desktop Object which is capable of holding other desktop objects (containers, applications, and documents). Since containers can contain other container objects, they can be used to implement hierarchical structures. Actions associated with this object are display contents, move into container, remove from container etc.

Attributes:

ConObjName

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Name of the container object

ContObjType

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

Type of container objects (for ex. directory)

IconForConObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Filename of the pixmap to be used as icon for the container object

IconForDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Filename of the pixmap to be used as icon for Desktop Object

NameOfDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Desktop Object Name

Operations:

int AcceptDrop (SrcName:char*, TrgName:char*)

Privilege: Public

No Inheritance

A container object accepts a drop of any other desktop objects.

void* GetAction

(Action_Name:char*, ConObjName:char*, ConObjType:int)

Privilege: Public

No Inheritance

Get the function which implements the specified action for container objects

```

int OpenConObj (ContObjName:char*)
    Privilege: Public
    No Inheritance
    Open the container object to show all the desktop objects contained in it.

int Remove (RemovePathName:char*)
    Privilege: Public
    No Inheritance
    Removes the container object if it does not contain any other desktop
    objects.

int CloseDtObj ()
    Privilege: Public
    Inherited From: CIDtDesktop_Object
    Close the Desktop object

int CopyDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)
    Privilege: Public
    Inherited From: CIDtDesktop_Object
    Copy the Source Desktop Object to the specified target

int DragDtObj (DtObjType:int)
    Privilege: Public
    Inherited From: CIDtDesktop_Object

void* GetAction
(Action_Name:char*,NameOfDtObj:char*,DtObjType:int)
    Privilege: Public
    Inherited From: CIDtDesktop_Object
    Get an action(s) from CIDtActions objects which is related to the
    desktop object of given type.

int MoveDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)
    Privilege: Public
    Inherited From: CIDtDesktop_Object
    Move the Source Desktop Object to the specified target

int OpenDtObj (DtObjType:int,IconForDtObj:char*)
    Privilege: Public
    Inherited From: CIDtDesktop_Object

```

5.6.2.4 Class CIDtDesktopWindow

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: CIDtDesktop_Object(Public) ISA - The Desktop window is a type of Desktop object.

Description:

The Desktop Window can be either in hierarchical format or iconic format. Desktop Window contains zero or more desktop objects. Desktop Window itself is a Desktop Object i.e., a container object.

Attributes:

DirectoryPath
Privilege: Private
Data Type: char*
Default Value: NOT IDENTIFIED
No Inheritance
Gives the Directory path of the Desktop Window

Operations:

void Refresh (WindowID:Widget, DirectoryPath:char*)
Privilege: Public
No Inheritance
Refresh the Desktop Window with the contents of the DirectoryPath in the specified Widget

5.6.2.5 Class CIDtDesktop_Object

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: CIDtDesktop_Object(Public) - A Desktop container object consists of other Desktop objects.
Class: CIDtActions(Public) HAS - A Desktop Object has default and other actions that can be performed on the object.
Class: CIDtDesktopWindow(Public) ISA - The Desktop window is a type of Desktop object.

Description:

Desktop object can be a container object, Application object or a document object. Each object will have zero or more related actions. Each desktop object will have an icon associated with it. A desktop object will be displayed using this icon.

Attributes:

IconForDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Filename of the pixmap to be used as icon for Desktop Object

NameOfDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Desktop Object Name

Operations:

int CloseDtObj ()

Privilege: Public

No Inheritance

Close the Desktop object

int CopyDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

No Inheritance

Copy the Source Desktop Object to the specified target

int DragDtObj (DtObjType:int)

Privilege: Public

No Inheritance

void* GetAction

(Action_Name:char*,NameOfDtObj:char*,DtObjType:int)

Privilege: Public

No Inheritance

Get an action(s) from CIDtActions objects which is related to the desktop object of given type.

int MoveDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

No Inheritance

Move the Source Desktop Object to the specified target

```
int OpenDtObj (DtObjType:int,IconForDtObj:char*)
```

Privilege: Public

No Inheritance

5.6.2.6 Class CIDtDocumentObject

Synopsis:

Parent Class: CIDtDesktop_Object

Is Not A Distributed Object

Is Associated With:

This class is derived from the class CIDtDesktop_Object

Description:

A type of Desktop object which provides a mechanism for associating multiple types of descriptive data with an object. Applications can be associated with a class of desktop document objects, and this application can be invoked to access the data described in the document object.

Attributes:

DocObjName

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Name of Document Object

DocObjType

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

Type of Document Object

IconForDocObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Filename of the pixmap to be used as icon for Document Object

IconForDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Filename of the pixmap to be used as icon for Desktop Object

NameOfDtObj

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktop_Object

Desktop Object Name

Operations:

void* GetAction

(Action_Name:char*,DocObjName:char*,DocObjType:int)

Privilege: Public

No Inheritance

Get the function which will be used to implement the specified action of the Document Object

char* GetAssociatedApplication (DocObjType:int)

Privilege: Public

No Inheritance

Every document object will have a related application for manipulating the data. This operation will be used to get the related application

int OpenDocObj (DocObjType:int,

CIDtApplicationObject.ApplName:char*)

Privilege: Public

No Inheritance

Open the Document object using the related application

int RemoveDocObject (DocObjName:char*)

Privilege: Public

No Inheritance

Document objects can be removed. Hence this operation will be used to remove the specified document object.

int CloseDtObj ()

Privilege: Public

Inherited From: CIDtDesktop_Object

Close the Desktop object

int CopyDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

Copy the Source Desktop Object to the specified target

int DragDtObj (DtObjType:int)

Privilege: Public

Inherited From: CIDtDesktop_Object

void* GetAction

(Action_Name:char*,NameOfDtObj:char*,DtObjType:int)

Privilege: Public

Inherited From: CIDtDesktop_Object

Get an action(s) from CIDtActions objects which is related to the desktop object of given type.

int MoveDtObj (SrcNameOfDtObj:char*, TrgNameOfDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

Move the Source Desktop Object to the specified target

int OpenDtObj (DtObjType:int,IconForDtObj:char*)

Privilege: Public

Inherited From: CIDtDesktop_Object

5.6.2.7 Class CIDtHierarchical

Synopsis:

Parent Class: CIDtDesktopWindow

Is Not A Distributed Object

Is Associated With:

This class is derived from the class CIDtDesktopWindow

Description:

Displays desktop objects in hierarchical format.

Attributes:

Tree

Privilege: Private

Data Type: Tree

Default Value: NOT IDENTIFIED

No Inheritance

A datastructure to represent the directory structure in a tree form.

DirectoryPath

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktopWindow

Gives the Directory path of the Desktop Window

Operations:

Tree CreateTree (S-Exp:char*)

Privilege: Public

No Inheritance

Creates and returns a Tree from a S-Exp.

int DisplayTree (Tree:Tree,WindowID:Widget)

Privilege: Public

No Inheritance

Displays a Tree(Directory structure) on the specified window.

int Refresh (Tree:Tree,WindowID:Widget)

Privilege: Public

No Inheritance

Refresh by displaying the specified Tree in the Window

void Refresh (WindowID:Widget, DirectoryPath:char*)

Privilege: Public

Inherited From: CIDtDesktopWindow

Refresh the Desktop Window with the contents of the DirectoryPath in the specified Widget

5.6.2.8 Class CIDtIconic

Synopsis:

Parent Class: CIDtDesktopWindow

Is Not A Distributed Object

Is Associated With:

This class is derived from the class CIDtDesktopWindow

Description:

Displays Desktop objects in Iconic form.

Attributes:

DirectoryPath

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

No Inheritance

Directory Path to be displayed in the desktop.

NumberOfObjects

Privilege: Private

Data Type: int

Default Value: 0

No Inheritance

Number of DesktopObjects in the Iconic Display

DirectoryPath

Privilege: Private

Data Type: char*

Default Value: NOT IDENTIFIED

Inherited From: CIDtDesktopWindow

Gives the Directory path of the Desktop Window

Operations:

```
int Refresh (WindowID:Widget, DirectoryPath:char*)
```

Privilege: Public

No Inheritance

Refresh by displaying the contents of the specified directory.

```
void Refresh (WindowID:Widget, DirectoryPath:char*)
```

Privilege: Public

Inherited From: CIDtDesktopWindow

Refresh the Desktop Window with the contents of the DirectoryPath in the specified Widget

5.7 Local Information Manager Classes

5.7.1 Local Information Manager Classes Overview

The LIMGR provides single-site search and access services. The LIMGR accepts queries and data access requests for execution. It acts as a search agent on behalf of users by identifying the sources of the data and transforming the search and access operations into requests which are acceptable to other data sources, such as V0 Gateways or Science Data Server. Clients interface with the LIMGR to determine the status of the search or to obtain the search results. The details of the underlying requests to the other agents are hidden from the user, except when the user requests these details. The LIMGR uses the information in the DDICT database in order to decompose the queries and determine the optimal request structure to satisfy the request. Science Data Servers make themselves available to LIMGRs by exporting information to the DIMGR which provides updates to the DDICT database.

The LIMGR uses the Server Request Framework key mechanism documented in 305-CD-028-002 to provide asynchronous request processing.

The LIMGR public classes provide the public classes for the DIMGR which is why no DIMGR public classes are documented in this section.

The DIMGR provides distributed search and access services. The DIMGR accepts queries and data access requests for execution. It acts as a search agent on behalf of users by identifying the sources of the data and transforming the search and access operations into requests which are acceptable to other data sources, such as Local Information Manager or Science Data Server. Users interface with the DIMGR to determine the status of the search or to obtain the search results. The details of the underlying requests to the other agents are hidden from the user, except when the user requests these details. The DIMGR uses the information in the DDICT database in order to decompose the queries and determine the optimal request structure to satisfy the request. LIMGRs and Science Data Servers make themselves available to DIMGRs by exporting information to the DIMGR which provides updates to the DDICT database.

The DIMGR uses the Server Request Framework key mechanism documented in 305-CD-028-002 to provide asynchronous request processing.

5.7.2 Local Information Manager Class Descriptions

5.7.2.1 Class DmImCIAdmRequestServer

Synopsis:

Parent Class: DmImCIRequestServer
 Distributed Object
 Is Associated With:
 Class: DmImMsgBase(Private) uses

Description:

This object is used for sending administration command to the server side and receiving information from the server. It is synchronously connected to the server .

Attributes:

myCommandType

Privilege: Public
 Data Type: enum DmElmCommandType
 Default Value: {NONE}
 No Inheritance
 Enumerated type which contains the command to be applied to the schemaList. The enumeration is {UPDATE, DELETE, ADD }

myRequestList

Privilege: Public
 Data Type: RWCollection
 Default Value: NOT IDENTIFIED
 No Inheritance
 ROGUE WAVE Collection that contains all the requests that are being tracked by the server.

mySchemaList

Privilege: Public

Data Type: GIParameterList

Default Value: NOT IDENTIFIED

No Inheritance

GIParameterList that contains the information to modify the schema of the LIM. It will be acted upon by myCommandType at the server side.

myStatus

Privilege: Public

Data Type: EcUtStatus

Default Value: NOT IDENTIFIED

No Inheritance

status flag returned after SetCommand and SetSchemaList .This enables the caller to check whether the functions performed their actions without errors.

myUserList

Privilege: Public

Data Type: RWCollection

Default Value: NOT IDENTIFIED

No Inheritance

Rogue Wave Collection of all the users that have a request pending at the server side.

myMsg

Privilege: Protected Attribute

Data Type: EcCsMsg *

Default Value: NOT IDENTIFIED

Inherited From: DmImCIRequestServer

Pointer to the message class that is used by DmImCIRequestServer to communicate and pass parameters to the server side.

myRequestServer

Privilege: Public

Data Type: EcUrUr

Default Value: NOT IDENTIFIED

Inherited From: DmImCIRequestServer

Contains the UR that identified the ServerRequest Object. Since there is one Server request object per client per server connection the ID of that object is used as a session ID between the caller and the server.

myServer

Privilege: Public

Data Type: EcUrUr

Default Value: NOT IDENTIFIED

Inherited From: DmImClRequestServer

Contains the UR that enables the serverRequest object to establish the connection with the server.

mySessionCommand

Privilege: Public

Data Type: enum DmEImSessionCommandType

Default Value: {RESTORE}

Inherited From: DmImClRequestServer

Enumerated type that contains the basic command that applies to session management. The default value is a RESTORE which is applied when the caller first connects to the server.

myStatus

Privilege: Public

Data Type: EcUtStatus

Default Value: NOT IDENTIFIED

Inherited From: DmImClRequestServer

Returned to the caller after a function call. Allows the caller to verify that function operated properly

myUser

Privilege: Public

Data Type: MSSUserProfile &

Default Value: NOT IDENTIFIED

Inherited From: DmImClRequestServer

This identifies for which user the DmImClrequestServer is created. That information will be used at the server side to check for permissions and to tag the request for each user.

Operations:

```
void DmImClAdmRequestServer (server :EcUrUr , user :  
MSSUserProfile &)
```

Privilege: Public

No Inheritance

DmEImCommandType GetCommand (void)

Privilege: Public

No Inheritance

This method will return the command type to the caller

GlpParameterList GetSchemaList (void)

Privilege: Public

No Inheritance

This method will return the SchemaList to the caller as a GlpParameterList

RWCollection & ListAllRequests (void)

Privilege: Public

No Inheritance

This method will return to the caller a reference to the collection of all the requests that are being tracked at the server.

RWCollection & ListAllUsers (void)

Privilege: Public

No Inheritance

This method will return to the caller a reference to the collection of all the users that have a pending request at the server.

EcUtStatus SetCommand (cmdType :DmEImCommandType)

Privilege: Public

No Inheritance

EcUtStatus SetSchemaList (SchemaList :GlpParameterList)

Privilege: Public

No Inheritance

this method will set the SchemaList provided by the caller into a local Glparameter List

EcUtStatus Submit (void)

Privilege: Public

No Inheritance

This method will check to see if CommandType are set and SchemaList is set then will create a new message object package the paramters and send that message to the server for processing. The message passing and processing is handle accoeding to the Server Request Framework Paragdim.

```
void ~DmImClAdmRequestServer ( )
```

Privilege: Public

No Inheritance

Basic destructor. when invoked all message objects and the connection to the server will be destroyed.

```
void DmImClRequestServer (server :EcUrUr, user  
:MSSUserProfile &)
```

Privilege: Public

Inherited From: DmImClRequestServer

This server class is inheriting from EcCsRequestServer . It is synchronously connected to the DmImSrRequestServer which server ID was provided at construction. It is used to manage the user session and to create objects capable of asynchronous communication between the client and server.

```
EcUtStatus NewRequest (request :DmImClrequest *, requestT  
:RequestType)
```

Privilege: Public

Inherited From: DmImClRequestServer

The caller initiates the creation of a newrequest using the DmImClRequest Server. It needs to pass a pointer to that object. That pointer will be assign to Null until the request is build. The pointer will then be assigned to that request and is ready for use. myRequestType will be used to differentiate Queries from Browse from Acquire. Eventually it may be possible by the use of typed constructors to have this step transparent to the caller but for the time being it is expected that different type of requests may use the same constructor but may need different overloaded methods later on. Therefore the type of request needs to be specified. NewRequest will build a message object DmImRequestMsg , encapsulate information about the request to be, and use the EcCsMsgHandler to ship that message to the DmImSrRequestServer. Note that this is a necessary step in creating an asynchronous request. The message object is nothing more than a courier which contains the information allowing both side to coordinate their efforts.

```
EcUtStatus SessionManager (mySessionCommand :  
DmEImSessionCommandtype)
```

Privilege: Public

Inherited From: DmImClRequestServer

`void ~DmImClRequestServer ()`

Privilege: Public

Inherited From: DmImClRequestServer

Disconnect from the server , All DmImClRequest Objects are destroyed. All Message objects available at the client side are destroyed.

5.7.2.2 Class DmImClRequest

Synopsis:

Parent Class: EcCSAsynchRequest_C

Distributed Object

Is Associated With:

Class: DmImMsgBase(Public) uses

Description:

This class will handle all requests submitted by the caller . It is part of the Server Request Framework by inheriting from EcCSAsynchRequest_C.

Attributes:

`myCallBack`

Privilege: Private

Data Type: DmImCallBack

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a pointer of type DmImCallback to a function provided by the caller.

`myCommandList`

Privilege: Public

Data Type: RWTPtrOrderedVector<DsClCommand>

Default Value: NOT IDENTIFIED

No Inheritance

Contains the collection of commands created by the caller.

`myMsg`

Privilege: Private

Data Type: EcCsMsg *

Default Value: NOT IDENTIFIED

No Inheritance

Points to the message class that is used by the request to communicate and pass parameters to the server side.

myRequestType

Privilege: Public

Data Type: enum RequestType

Default Value: NOT IDENTIFIED

No Inheritance

Caller specified which request type he is using. That information will be needed to properly process the parameters that are associated with the request

myStatus

Privilege: Public

Data Type: EcUtStatus

Default Value: NOT IDENTIFIED

No Inheritance

Contains status information returned to the caller when calling a method

myUR

Privilege: Public

Data Type: EcUrUr

Default Value: NOT IDENTIFIED

No Inheritance

Contains the Unique reference of the request. This information needs to be available to the caller to allow the caller to differentiate a request from another one.

Operations:

EcUtStatus AddCommand (oneCmd :DsClCommand)

Privilege: Public

No Inheritance

Will allow the caller to add a command to the the list of commands contained in myCommandList. The entire set is passed to the message class when submit is called.

void DmImClRequest (DmImSrRequest &)

Privilege: Public

No Inheritance

The construction of DmImClRequest is made at the server side at the time DmImSrRequest is being created. All the initialization informations are extracted from the server side object. That is why the argument for the constructor of DmImClRequest is a reference to the server side request DmImSrRequest.

`int EstimateTimeUse (void)`

Privilege: Public

No Inheritance

This method will send a special query to the server side to request an estimate on the time expected to be used to complete the query. The information will be packaged with the query and the return will be made through the same message object. The number returned correspond to the number of seconds.

`EctChar* GetBrowseResults (startpoint :int , endpoint :int)`

Privilege: Public

No Inheritance

This method will allow the caller to retrieve search results . Specified are the startpoint and the endpoint which allow for a range of results to be returned. This allows to customize the size of the results to the caller's workstation capabilities. The type of result returned is EctChar *.

`RWTPtrOrderedVector<DsClCommand> GetCommands (void)`

Privilege: Public

No Inheritance

This method will be used when the caller needs to retrieve from the query a set of commands following the DsClCommandBase paradigm.

`GlpParameterList GetResults (startpoint :int, endpoint :int)`

Privilege: Public

No Inheritance

This method will allow the caller to retrieve search results . Specified are the startpoint and the endpoint which allow for a range of results to be returned. This allows callers to customize the size of the results to the caller's workstation capabilities.

`EcUrUr GetUR (void)`

Privilege: Public

No Inheritance

This method will allow the caller to get a DmImClrequest UR returned. It will be used to identify one request from another.

`void SetCallBack (DmImCallBack *)`

Privilege: Public

No Inheritance

Allows the caller that created the request to be notified when a state change happens within the request. A state change for example will occur when the request has completed its query and the results are available. The function supplied has to accept two parameters, myUR : EcUrUr and myState, an enumerated type inherited from SRF. myUR will be used by the caller to identify which request is calling back.

`EcUtStatus SetRequestType (newrequestType :RequestType)`

Privilege: Public

No Inheritance

Will allow the caller to change the request type. This is usefull when the same request is being updated to provide additional services . As an example a BROWSE request would be updated to become an ACQUIRE request.

`EcUtStatus SetSearchConstraints (constraints
:GlParameterList)`

Privilege: Public

No Inheritance

This method will accept a GlParameterlist as argument and pass this argument to the message class . It returns a status of the operation. The Message class is used within the SRF for asynchronous communication between the caller and the Server.

`EcUtStatus SetSearchConstraints (constraints :RWCString)`

Privilege: Public

No Inheritance

This method will accept RWCString as search constraint and pass the argument to the message class. It returns a status of the operation.

`void StateChange (void)`

Privilege: Private

No Inheritance

This method will set the correct state for the request object and will invoke the function supplied by the caller and referred by the pointer DmImCallback and pass two parameters. myUR :EcUrUr which identifies the request and the state : an enumerated type inherited from SRF that is used to reveal the state of the request

EcUtStatus Submit (void)

Privilege: Public

No Inheritance

When caller invokes submit, the request will finish to encapsulate all commands or constraints into a message object and ship that message object to the server side where it will be processed. The message object is determined by the request type and it is shipped using the EcCsMsgHandler object (SRF).

void ~DmImClRequest ()

Privilege: Public

No Inheritance

When called the request and the message associated with it will be destroyed.

5.7.2.3 Class DmImClRequestServer

Synopsis:

Parent Class: EcCsRequestServer_C

Distributed Object

Is Associated With:

Class: DmImMsgBase/Private) uses

Description:

This object is used for creating requests and to send session command to the server side. It is synchronously connected to the server and its UR is used as a session Id.

Attributes:

myMsg

Privilege: Protected Attribute

Data Type: EcCsMsg *

Default Value: NOT IDENTIFIED

No Inheritance

Pointer to the message class that is used by DmImClRequestServer to communicate and pass parameters to the server side.

myRequestServer

Privilege: Public

Data Type: EcUrUr

Default Value: NOT IDENTIFIED

No Inheritance

Contains the UR that identified the ServerRequest Object. Since there is one Server request object per client per server connection the ID of that object is used as a session ID between the caller and the server.

myServer

Privilege: Public

Data Type: EcUrUr

Default Value: NOT IDENTIFIED

No Inheritance

Contains the UR that enables the serverRequest object to establish the connection with the server.

mySessionCommand

Privilege: Public

Data Type: enum DmEImSessionCommandType

Default Value: {RESTORE}

No Inheritance

Enumerated type that contains the basic command that applies to session management. The default value is a RESTORE which is applied when the caller first connects to the server.

myStatus

Privilege: Public

Data Type: EcUtStatus

Default Value: NOT IDENTIFIED

No Inheritance

Returned to the caller after a function call. Allows the caller to verify that function operated properly

myUser

Privilege: Public

Data Type: MSSUserProfile &

Default Value: NOT IDENTIFIED

No Inheritance

This identifies for which user the DmImClrequestServer is created. That information will be used at the server side to check for permissions and to tag the request for each user.

Operations:

```
void DmImClRequestServer (server :EcUrUr, user
:MSSUserProfile &)
```

Privilege: Public

No Inheritance

This server class is inheriting from EcCsRequestServer . It is synchronously connected to the DmImSrRequestServer which server ID was provided at construction. It is used to manage the user session and to create objects capable of asynchronous communication between the client and server.

```
EcUtStatus NewRequest (request :DmImClrequest *, requestT
:RequestType)
```

Privilege: Public

No Inheritance

The caller initiates the creation of a newrequest using the DmImClRequest Server. It needs to pass a pointer to that object. That pointer will be assign to Null until the request is build. The pointer will then be assigned to that request and is ready for use. myRequestType will be used to differentiate Queries from Browse from Acquire. Eventually it may be possible by the use of typed constructors to have this step transparent to the caller but for the time being it is expected that different type of requests may use the same constructor but may need different overloaded methods later on. Therefore the type of request needs to be specified. NewRequest will build a message object DmImRequestMsg , encapsulate information about the request to be, and use the EcCsMsgHandler to ship that message to the DmImSrRequestServer. Note that this is a necessary step in creating an asynchronous request. The message object is nothing more than a courier which contains the information allowing both side to coordinate their efforts.

```
EcUtStatus SessionManager (mySessionCommand :
DmEImSessionCommandtype)
```

Privilege: Public

No Inheritance

```
void ~DmImClRequestServer ()
```

Privilege: Public

No Inheritance

Disconnect from the server , All DmImClRequest Objects are destroyed. All Message objects available at the client side are destroyed.

5.8 Event Handling Classes

5.8.1 Event Handling Classes Overview

Events, errors, faults need a uniform method of handling across all subsystems within ECS. It is important to provide common or public interface classes in conjunction with MSS provided services for the handling of faults and notification of events.

The generic status/error class is EcUtStatus, which is used as the standard return type for any public operation. EcUtStatus is one of the classes used in event handing found in the common software. Other event handling public classes are either have been incorporated in a framework developed by CSS or MSS, namely EcUtLogger, EcAgManager, EcAgEvent, EcAgHostInfo, EcAgTuble, EcAgMetric, EcAgPerfMetric, EcAgFaultMetric, EcAgConfigMetric, EcPfClient, and EcPfManagedServer. The class presented in this section is a generic class that is not embedded in any framework.

5.8.2 Error Handling Class Descriptions

5.8.2.1 Class EcUtStatus

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:

Description:

This class is a generic status/error return class. It should be used as the standard return type for any public operation. EcUtStatus includes a set of enumerated types for the most common status returns. A detailed status code field is provided so that the called operation can include more detailed information than the common status types. EcUtStatus can also be used as an argument for a function that returns a pointer or a value.

EcUtStatus contains an enumerated list of the most common return codes. However, sometimes a routine may need to give a more detailed status. In this case the routine should provide an enumerated list of detailed status codes, and should use one of the values as the detailed code in EcUtStatus.

Attributes:

Operations:

```
void EcUtStatus (void)
```

Privilege: Public

No Inheritance

Default constructor for EcUtStatus. Sets status code to OK

```
void EcUtStatus (status: EcUtStatusCode, detailed=0:EcTInt)
```

Privilege: Public

No Inheritance

Constructor used to initialize EcUtStatus to one of the predefined status codes.

```
boolean OK (status:EcUtStatusCode, detailed=0:EcTInt)
```

Privilege: Public

No Inheritance

Returns TRUE if the operation completed ok. Returns FALSE if the operation failed.

`boolean Failed (void)`

Privilege: Public

No Inheritance

Returns FALSE if the operation completed ok. Returns TRUE if the operation failed.

`void SetOk (void)`

Privilege: Public

No Inheritance

Sets the status state of the object to ok.

`void SetFailed (void)`

Privilege: Public

No Inheritance

Sets the state of the object to failed.

`void SetStatusCode (EcUtStatusCode)`

Privilege: Public

No Inheritance

Sets the status state of the object to one of the enumerated status codes.

`EcUtStatusCode GetStatusCode (void)`

Privilege: Public

No Inheritance

Gets the state of the object. The returned values is one of the enumerated status codes.

`void SetLogMessageLink (EcLgMsgLink)`

Privilege: Public

No Inheritance

Sets the link to an error message that was logged by the called routine. This link must be included in any error logging that is performed by the caller as a result of the failed operation.

`EcLgMsgLink GetLogMessageLink (void)`

Privilege: Public

No Inheritance

Returns the value of the link to an error message that was logged by the called routine.

```
void SetDetailed (EcTInt)
```

Privilege: Public

No Inheritance

Sets the value of the detailed error code. The detailed error code can be used by a routine to provide more detailed status codes than are provided in the EcUtStatusCode enumerated types. NOTE: It is the responsibility of the called routine to document the values of the detailed codes in the header file which declares the operation.

```
EcTInt GetDetailed (void)
```

Privilege: Public

No Inheritance

Gets the value of the detailed error code. The detailed error code is used by a called routine to provide more detailed status codes than are provided in the EcUtStatusCode enumerated types. NOTE: It is the responsibility of the called routine to document the values of the detailed codes in the header file which declares the operation.

5.9 Version 0 Gateway Classes

5.9.1 Version 0 Gateway Classes Overview

Gateway provides interoperability with V0 for directory level queries, inventory level queries, browse requests and product orders. Version 0 queries originating from Version 0 clients will be sent to a V0/ECS gateway which will operate at each DAAC, and emulate a Version 0 IMS server. The gateway will translate a incoming V0 ODL request into ECS query language and submit it to the local ECS data server. The result will be returned to the Gateway, which then will reformat it into V0 ODL structures and return it to V0 client. The Gateway uses a database constructed by a gateway administrator using the V0 search parameters, ECS schema and metadata. The Advertising Service (ADSRV) CSCI and Document Data Server (DDSRV) CSCI make use of the gateway database to resolve ECS to V0 mappings.

In Release B, the Gateway provides interoperability between the ECS clients and the V0 IMS servers at the DAACs. The public interface to the Gateway will be the same as specified in the Distributed Information Manager CSCI. No additional classes are shown here for the public interface to the Gateway.

5.9.2 Version 0 Gateway Class Descriptions

5.9.2.1 Class DmGwAcquireRequest

Synopsis:

Parent Class: DsClRequest

Is Not A Distributed Object

Is Associated With:

This class is derived from the class DsClRequest

Description:

This class contains all the information and the operations required to submit a product ordering request to the data server.

Attributes:

myCollector

Privilege: Private

Data Type: DsCIESDTRReferenceCollector *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the universal reference to the data server which the data collection being requested is binded to.

Operations:

GlStatus Abort ()

Privilege: Public

No Inheritance

This operation cancels the product acquisition request that is being submitted.

Void AcquireComplete (whatHappened:GlStatus)

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer after the product acquisition request submitted to the data server is completed successfully.

Void AcquireFailed ()

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer if a product acquisition request submitted to the data server can not be completed due to the reason specified in the argument whatHappened.

Void DmGwAcquireRequest (dataServerUR:GlUR &, dataURs:URVec
&, pullerInfo:UserInfo &, deliveryMech:DmGwMediaInfo &)

Privilege: Public

No Inheritance

This operation constructs a product ordering request.

GlStatus Status ()

Privilege: Public

No Inheritance

This operation is invoked if a desire to check the status of product acquisition is required.

GlStatus Submit ()

Privilege: Public

No Inheritance

This operation is invoked to submit the product acquisition request to the data server.

Void ~DmGwAcquireRequest ()

Privilege: Public

No Inheritance

This operation destroys the structure of product ordering request.

5.9.2.2 Class DmGwBrowseRequest

Synopsis:

Parent Class: DsClRequest

Is Not A Distributed Object

Is Associated With:

This class is derived from the class DsClRequest

Description:

This class contains all the information and the operations required to submit a browse request to the data server.

Attributes:

myCollector

Privilege: Private

Data Type: DsClESDTRreferenceCollector *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the universal reference to the data server which the browse image being requested is binded to.

Operations:

GlStatus Abort ()

Privilege: Public

No Inheritance

This operation cancels the browse image request that is being submitted.

Void BrowseComplete (whatHappened:GlStatus)

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer after a browse request submitted to the data server is completed successfully.

Void BrowseFailed ()

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer if a browse request submitted to the data server can not be completed due to the reason specified in the argument whatHappened.

Void DmGwBrowseRequest (dataServer:Glur &
outputStreamVector:StreamVec &, URVector:URVec &)

Privilege: Public

No Inheritance

This operation constructs a browse image request.

GlStatus Status ()

Privilege: Public

No Inheritance

This operation is invoked if a desire to check the status of browse request is required.

GlStatus Submit ()

Privilege: Public

No Inheritance

This operation is invoked to submit the browse image request to the data server.

Void ~DmGwBrowseRequest ()

Privilege: Public

No Inheritance

This operation destroys the structure of browse image request.

5.9.2.3 Class DmGwDistribution

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: DmGwMediaInfo(Public) contains - This association indicates that the DmGwDistribution contains a collection of DmGwMediaInfo.

Description:

This class contains all the information and the operations required to acquire product distribution and format information.

Attributes:

myDSSUniversalReference

Privilege: Private

Data Type: GIUR &

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the universal reference to the distribution data server.

Operations:

Void DmGwDistribution (dataServer:GIUR &)

Privilege: Public

No Inheritance

This operation constructs a distribution request for the product format information to the distribution data server.

DmGwMediaInfo * getFirstMediaInfo ()

Privilege: Public

No Inheritance

This operation retrieves the information of first media format available for the product being ordered.

DmGwMediaInfo * getNextMediaInfo ()

Privilege: Public

No Inheritance

This operation retrieves the information of next media format available for the product being ordered.

int numberOfMediaInfo ()

Privilege: Public

No Inheritance

This operation retrieves the number of different media formats for the product being ordered.

Void ~DmGwDistribution ()

Privilege: Public

No Inheritance

This operation destroys the structure of the distribution request for the product format information.

5.9.2.4 Class DmGwGateWayCollector

Synopsis:

Parent Class: DsCIESDTRReferenceCollector

Is Not A Distributed Object

Is Associated With:

Class: DmGwGateWayDescriptor(Public) myValid - This association indicates that DmGwGateWayCollector is a collection of DmGwGateWayDescriptor valids.

Description:

This class contains all the information and the operations required to retrieve a collection of valids exported from the data server.

Attributes:

Operations:

Void DmGwGateWayCollector ()

Privilege: Public

No Inheritance

This operation constructs a valid export request to the data server.

DmGwGateWayDescriptor * getFirstDescriptor ()

Privilege: Public

No Inheritance

This operation retrieves the information of the first valid descriptor exported from the data server.

DmGwGateWayDescriptor * getNextDescriptor ()

Privilege: Public

No Inheritance

This operation retrieves the information of the next valid descriptor exported from the data server.

int getNumberOfDescriptors ()

Privilege: Public

No Inheritance

This operation retrieves the number of valid descriptors exported from the data server.

Void ~DmGwGateWayCollector ()

Privilege: Public

No Inheritance

This operation destroys the structure of the valid export request.

5.9.2.5 Class DmGwGateWayDescriptor

Synopsis:

Parent Class: DsCIDescriptor

Is Not A Distributed Object

Is Associated With:

Class: DmGwGateWayCollector(Public) myValid - This association indicates that DmGwGateWayCollector is a collection of DmGwGateWayDescriptor valids.

Description:

This class contains the valid information which is returned by the data server after the valid export request is completed.

Attributes:

Operations:

Void DmGwGateWayDescriptor ()

Privilege: Public

No Inheritance

This operation constructs a valid descriptor for storing exported valid information.

StringVec getAccessRestrictions ()

Privilege: Public

No Inheritance

This operation returns the ordering restriction and legal prerequisites placed on the data set.

DmGwSpatial & getBoundingRectangle ()

Privilege: Public

No Inheritance

This operation returns the specification of the spatial coverage for each data set.

DmGwGateWayDescriptor & getBrowseDescriptor ()

Privilege: Public

No Inheritance

This operation returns the browse descriptor, if exists, which is related to the data set.

StringVec getCampaignName ()

Privilege: Public

No Inheritance

This operation returns the name of campaign or project that gathered the data set.

StringVec getDataCollectionDescription ()

Privilege: Public

No Inheritance

This operation returns the major emphasis of the content of the data collection.

DataCollectionId getDataCollectionName ()

Privilege: Public

No Inheritance

This operation returns the recommended name to be used when referring to this data set.

DayNight getDayNightIndicator ()

Privilege: Public

No Inheritance

This operation returns the day/night indicator for the data set.

DifEntryId getDiffEntryId ()

Privilege: Public

No Inheritance

This operation returns the entry id of the Global Change Master Directory for the data set.

GeoPhysParamList & getGeoPhysicalParams ()

Privilege: Public

No Inheritance

This operation returns the specification of the geophysical parameters referenced in the data set.

StringVec getInstrumentNames ()

Privilege: Public

No Inheritance

This operation returns the abbreviation, acronym, or other common name of the instrument sensor by which the data set is collected.

String getLocalityName ()

Privilege: Public

No Inheritance

This operation returns the spacial coverage described for the data set.

ProcessingLevel getProcessingLevel ()

Privilege: Public

No Inheritance

This operation returns the classification of the science data processing level which defines the characteristics of the data set.

RangeDateTime getRangeDateTime ()

Privilege: Public

No Inheritance

This operation returns the temporal coverage period extended for the data set.

StringVec getSatelliteNames ()

Privilege: Public

No Inheritance

This operation returns the name of the satellite on which the data set is collected.

StringVec pairingOfInstrumentSatellite ()

Privilege: Public

No Inheritance

This operation returns the matching pairs for instrument sensor and satellite.

Void ~DmGwGateWayDescriptor ()

Privilege: Public

No Inheritance

This operation destroys the structure of valid descriptor.

5.9.2.6 Class DmGwInvESDTRreference

Synopsis:

Parent Class: DsCIESDTRreference

Is Not A Distributed Object

Is Associated With:

Class: DmGwInvSearchRequest(Public) GranuleProduct - This association indicates that the DmGwInvESDTRreference is a collection of granule product returned from the inventory search request DmGwInvSearchRequest.

Description:

This class contains the information of granule references returned from an inventory search request for a particular dataset.

Attributes:**Operations:**

`Void DmGwInvESDTReference ()`

Privilege: Public

No Inheritance

This operation constructs the structure for storing granule related information.

`GIUR & getBrowse ()`

Privilege: Public

No Inheritance

This operation retrieves the browse reference, if exists, which is related to the individual granule returned in the inventory data set.

`DayNight getDayNightFlag ()`

Privilege: Public

No Inheritance

This operation retrieves the day/night indication of individual granule returned for the inventory data set.

`DateTime getEndDate ()`

Privilege: Public

No Inheritance

This operation retrieves the ending time of the temporal coverage for the individual granule returned in the inventory data set.

`DmGwSpatial & getSpatialExtent ()`

Privilege: Public

No Inheritance

This operation retrieves the spatial coverage of individual granule returned in the inventory data set.

`DateTime getStartDate ()`

Privilege: Public

No Inheritance

This operation retrieves the starting time of the temporal coverage for the individual granule returned in the inventory data set.

GIUR & getUR ()

Privilege: Public

No Inheritance

This operation returns the universal reference to the individual granule returned in the inventory data set.

Void ~DmGwInvESDTRreference ()

Privilege: Public

No Inheritance

This operation destroys the granule reference structure.

5.9.2.7 Class DmGwInvQuery

Synopsis:

Parent Class: DsClQuery

Is Not A Distributed Object

Is Associated With:

Class: DmGwInvSearchRequest(Public) myInvQuery - This association indicates that DmGwInvQuery is the query criteria when the inventory search request DmGwInvSearchRequest is submitted.

Description:

This class contains the information of the query criteria for the data set requested.

Attributes:

Operations:

```
Void DmGwInvQuery (dataCollection:DataCollectionId &,
spatialConstraint:DmGwSpatial &,
temporalConstraint:DmGwTemporal &,
procLevelConstraint:ProcessingLevel &,
dayNightConstraint:DayNight &,
geoPhysicalParams:GeoPhysParamList &, granuleLimit:int)
```

Privilege: Public

No Inheritance

This class contains the information of the query criteria for the data set requested.

Void ~DmGwInvQuery ()

Privilege: Public

No Inheritance

This operation destroys the query structure for the data set.

5.9.2.8 Class DmGwInvSearchRequest

Synopsis:

Parent Class: DsCIESDTRedirectCollector

Is Not A Distributed Object

Is Associated With:

Class: DmGwInvESDTRedirect(Public) GranuleProduct - This association indicates that the DmGwInvESDTRedirect is a collection of granule product returned from the inventory search request DmGwInvSearchRequest.

Class: DmGwInvQuery(Public) myInvQuery - This association indicates that DmGwInvQuery is the query criteria when the inventory search request DmGwInvSearchRequest is submitted.

Description:

This class contains all the information and the operations required to submit an inventory search request to the data server.

Attributes:

Operations:

GlStatus Abort ()

Privilege: Public

No Inheritance

This operation cancels the inventory search request that is being submitted.

Void DmGwInvSearchRequest (dataServer:GlUR &
endUser:DmGwUserInfo &, searchConstraint:DmGwInvQuery &)

Privilege: Public

No Inheritance

This operation constructs an inventory search request.

Void InvSearchComplete ()

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer after the inventory search request submitted to the data server is completed successfully.

Void InvSearchFailed (whatHappened:GlStatus)

Privilege: Public

No Inheritance

This operation should be invoked by the communication layer if an inventory search request submitted to the data server can not be completed due to the reason specified in the argument whatHappened.

`GlStatus Status ()`

Privilege: Public

No Inheritance

This operation is invoked if a desire to check the status of inventory search request is required.

`GlStatus Submit ()`

Privilege: Public

No Inheritance

This operation is invoked to submit the inventory search request to the data server.

`DmGwInvESDTRreference * getFirstInvESDTRef ()`

Privilege: Public

No Inheritance

This operation retrieves the information of the first granule reference returned from the inventory search request.

`DmGwInvESDTRreference * getNextInvESDTRef ()`

Privilege: Public

No Inheritance

This operation retrieves the information of the next granule reference returned from an inventory search request.

`int numberOfInvESDTRefs ()`

Privilege: Public

No Inheritance

This operation retrieves the number of granule references returned in the data set requested.

`Void ~DmGwInvSearchRequest ()`

Privilege: Public

No Inheritance

This operation destroys the structure of inventory search request.

5.9.2.9 Class DmGwMediaInfo

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: DmGwDistribution(Public) contains - This association indicates that the DmGwDistribution contains a collection of DmGwMediaInfo.

Description:

This class contains all the information and the operations required for acquiring distribution media format.

Attributes:**Operations:**

```
String getMediaFormat ()
```

Privilege: Public

No Inheritance

This operation retrieves the format of the distribution media.

```
String getMediaName ()
```

Privilege: Public

No Inheritance

This operation retrieves the name of the distribution media.

5.9.2.10 Class DsCIDDescriptor**Synopsis:**

No Parent Class

Distributed Object

Is Associated With:

None

Description:

The DsCIDDescriptor class is the proxy class which is imported from the Data Server Subsystem. The description of the server class is defined in the DID305 Data Server Subsystem section.

Attributes:**Operations:****5.9.2.11 Class DsCIESDTRreference****Synopsis:**

No Parent Class

Distributed Object

Is Associated With:

None

Description:

The DsCIESDTRreference class is the proxy class which is imported from the Data Server Subsystem. The description of the server class is defined in the DID305 Data Server Subsystem section.

Attributes:

Operations:

5.9.2.12 Class DsCIESDTReduceCollector

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
None

Description:

The DsCIESDTReduceCollector class is the proxy class which is imported from the Data Server Subsystem. The description of the server class is defined in the DID305 Data Server Subsystem section.

Attributes:

Operations:

5.9.2.13 Class DsCIQuery

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
None

Description:

The DsCIQuery class is the proxy class which is imported from the Data Server Subsystem. The description of the server class is defined in the DID305 Data Server Subsystem section.

Attributes:

Operations:

5.9.2.14 Class DsCIRequest

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
None

Description:

The DsCIESDTReduceRequest class is the proxy class which is imported from the Data Server Subsystem. The description of the server class is defined in the DID305 Data Server Subsystem section.

Attributes:

Operations:

5.10 Ingest Classes

5.10.1 Ingest Classes Overview

The Ingest Subsystem contains a collection of hardware and software that supports the ingest of data into ECS repositories on a routine and ad hoc basis and triggers subsequent archiving and/or processing of the data. The Ingest Subsystem configuration must be flexible to support a variety of data formats and structures, external interfaces, and ad-hoc ingest tasks. Data processing and storage functions to be performed by the Ingest Subsystem and ingest clients vary according to attributes of the ingested data such as data type, data format, and the level to which the ingested data has been processed. The ingest clients provide a single virtual interface point for the receipt of all external data to be archived within the SDPS. Individual ingest clients are established to support each unique interface, allowing the interface parameters to be modified as interface and mission requirements evolve. Ingest data preprocessing, metadata validation, and metadata extraction is performed by the ingest clients on any incoming data, as required. Data is staged to one of two areas depending on the data level, data type, and other data set specific characteristics. Level 0 data from ongoing missions will be staged to the Ingest Subsystem working storage area, where the data will be ingested and stored in the Level 0 rolling store. The staged data will also be accessible by the SDPS Processing Subsystem for that data which must be processed to higher levels. Level 1A-4 data will be staged directly to the working storage area in the Data Server Subsystem. Ingest client functionality such as quality checking and reading of metadata will be performed on this data upon the Data Server Subsystem processor hardware. The data server will then archive the data in the logical and physical data server to which the particular data has been assigned. From a software perspective, the Ingest Subsystem is organized into a collection of tools from which those required for a specific situation can be configured. The resultant configuration is called an ingest client. Ingest clients may exist in a static configuration to service a routine external interface, or they may be specially configured and exist only for the duration of a specific ad hoc ingest task. The hardware components of the Ingest Subsystem are similar to those of the data server, but are specialized to meet the ingest requirements at a given site. Specialized forms of ingest clients may be incorporated into site unique architectures, and additional processing hardware may also be incorporated at those sites where special transformations must be accomplished on ingest data sets.

5.10.2 Ingest Class Descriptions

5.10.2.1 Class InServerExtRPC_C

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

Class: CsGateway(Private) IsInvokedBy - The Gateway object interfaces with the InServerExtRPC to initiate a new Ingest Session through the Ingest Server.

InServer (Aggregation)

Description:

This is the client/proxy implementation that defines the RPC for initiating an Ingest Session.

Attributes:**Operations:**

```
int CreateSession (handle_t InServerBH, char
*GatewayStringBH, error_status_t *CreateSessStatus)
```

Privilege: Public

No Inheritance

This is a RPC that initiates a new Ingest Session.

5.10.2.2 Class InServerIntRPC_C**Synopsis:**

No Parent Class

Distributed Object

Is Associated With:

Class: InSession(Private) IsInvokedBy - InSession interfaces with InServerIntRPC_C to delete itself from the InServer's session list.

InServer (Aggregation)

Description:

This is the client/proxy implementation for the InServer object class. The provided services are to be used by the InSession object class.

Attributes:**Operations:**

```
void DeleteSession (handle_t InServerBH, int SessionId,
error_status_t *DelSessStatus)
```

Privilege: Public

No Inheritance

Deletes the specified session from the InServer's Session List.

5.10.2.3 Class InSessionEcsRPC_C**Synopsis:**

No Parent Class

Distributed Object

Is Associated With:

Class: InSession(Private) IsInvokedBy

CsGateWay (Aggregation)

Description:

This is the client/proxy implementation that defines services for sending outgoing data messages from the ECS Ingest.

Attributes:**Operations:**

```
void ecsDDN (handle_t GatewayBH, char *DDN, error_status_t  
ecsDDNstatus)
```

Privilege: Public

No Inheritance

The RPC is invoked by ECS Ingest to send the Data Delivery Notice (DDN) data message to the external client.

5.10.2.4 Class InSessionExtRPC_C**Synopsis:**

No Parent Class

Distributed Object

Is Associated With:

Class: CsGateWay(Public) Invokes - The Gateway object interfaces with the InSessionExtRPC object to deliver the DAN and DDA data messages received from the external Client to Ingest Session.

InSession (Aggregation)

Description:

This is the client/proxy implementation that defines the RPC (Remote Procedure Call) for delivering data message from the external Client to ECS/Ingest.

Attributes:**Operations:**

```
void extDAN (handle_t InSessBH, char *DANmsg, char **DAmsg,  
error_status_t *extDANstatus)
```

Privilege: Public

No Inheritance

This is the RPC that delivers the Data Availability Notice (DAN) data message from the external client to ECS Ingest.

```
void extDDA (handle_t InSessBH, char *DDAmsg, error_status_t  
*status)
```

Privilege: Public

No Inheritance

The is the RPC that delivers the Data Delivery Ack (DDA) from the external client to ECS Ingest.

5.10.2.5 Class InSessionIntRPC_C

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
InSession (Aggregation)

Description:

This is the client/proxy implementation for exporting the data messages to the InSession object class.

Attributes:

Operations:

```
void IntDDN (handle_t InSessBH, char *DDN, error_status_t  
IntDDNstatus)
```

Privilege: Public

No Inheritance

This is the RPC that exports the Data Delivery Notice (DDN) data message to the InSession object class.

5.11 Management Subsystem Classes

5.11.1 Management Subsystem Classes Overview

Fault Management: Fault Management addresses the detection, diagnosis, isolation and resolution of faults associated with the managed objects within ECS. The managed objects comprise networks, hosts and applications. A fault is an unacceptable change in the state of a managed object. Fault management provides for the detection of changes in state of managed objects in order to be able to distinguish the unacceptable changes that constitute faults from acceptable changes. Fault Management, therefore, in conjunction with the mode management service, provides the capabilities for real-time configuration management to include the startup, shutdown and discovery of ECS applications. Further, since the service maintains the status of resources, it provides the capability to provide the status of these resources, such as processors and associated disks, upon requests from subsystems such as the Planning Subsystem.

Accountability Management: The Accountability Management Service provides the capabilities of User Registration and the generation of reports from audit trails. ECS provides for two generic classes of users: guest users and registered users. Guest users are users that have not formally registered to become registered users. Registered users are those users that have submitted requests for a registered user account, and have had accounts created for them, based on an approval process. registered users are allowed access to services and products beyond those available to guest users. User registration provides the operators the capability to create accounts against requests submitted by guest users wishing to become authorized ECS users. The registration service provides the capabilities for the creation, modification and maintenance of accounts with user profiles. The user profile information contains user identification, user class, field(s) of research, investigating group affiliation (if any), shipping address, and electronic mail

address (if any). The Accountability Management service makes the user profile available to the various subsystems, such as Data Server subsystem for information such as the user's electronic mail address and the shipping address, used for the distribution of data products ordered. The Audit Trail capability provides the means to verify the integrity of the system. This comprises the generation of a user audit trail and a security audit trail with data collected from a variety of sources.

Management Agent Services: The MSS provides ECS M&O Staff with the capability to manage the ECS enterprise, i.e., to perform network and system management services on all ECS resources, including all SDPS, FOS, and CSMS components. Mode Management monitoring and control capabilities are enabled through the use of agent services as well. The enterprise management system is based on the manager-agent model. It consists of management applications, a managed object model, a management protocol, and server objects for RPC communications. The management applications reside on managing system(s). They provide the interfaces for the human enterprise manager to perform management tasks. The managed object model consists of managed objects which are defined to represent the resources being managed. The underlying resources can be physical devices, system software or applications. The management agent is the implementation which substantiates the managed objects. It normally resides on each remote host performing monitoring and control functions for the management applications which are on the managing system(s). The management applications communicate with agents through the management protocol. The MSS is composed of a variety of management applications providing services such as fault, performance, mode, security, and accountability management for ECS networks, hosts, as well as SDPS and FOS applications. The management applications reside on MSS Server. The management information of remote objects need to be conveyed to the management applications through the Management Agent Service which primarily resides on remote hosts.

The following functionality is new to Release B:

Billing and Accounting: ECS operations are supported by integrated and automated billing and accounting functions. The Enterprise Monitoring and Coordination (EMC) Billing and Accounting Application Service (BAAS) provides the mechanisms for ECS to price user data transactions, invoice users for system usage, and meet ECS' needs to track and to provide financial data. The necessary flow of information between the functions provided by the BAAS makes it unlikely that any one function can successfully stand alone or operate independently. More importantly, the BAAS depends on information provided by other sub-systems and external entities. Science user accounts, for example, are set up in BAAS using information received from the Accountability Management Service. One of the BAAS' primary function is to provide bill-back capabilities. The billing functionality allows ECS to gather and track information on science user data orders, and to price these data orders based on different costed resources (e.g., disk utilization, CPU, media, connect time) or standard product ordered using pricing algorithms associated with each product. For purposes of estimating the price of a new product request, pricing algorithms maintained in pricing tables in BAAS will be made available to the DSS. A standard pricing policy is assumed for ECS products. The DPS and DSS will provide the BAAS with accounting and resource data for science user orders which have been fulfilled. The invoicing functionality allows ECS to inform accounts of their total activity during a particular billing cycle and of the charges associated with such activity. Statements will be generated monthly (and be configurable to another length

of time) and will either take the form of a statement of account or of a bill invoice. Policy may dictate that no charges be applied to any account, or to particular accounts; or that certain accounts be measured on resources consumed (e.g., number of tapes, number of images) rather than dollars. In such cases, the accounts would not receive a bill invoice but a statement of account. Unlike bill invoices, the statements of account do not anticipate payment. An account also would receive a statement of account instead of a bill invoice when the account has funds credited to it in advance of data orders. As charges are incurred by the account, these are deducted (debited) from the existing credits. The statements of account would show activity and balance remaining. An account's balance status also will be available for on-line consulting via the CLS. Science user payments (made in the form of checks or purchase orders) will be credited to the appropriate accounts and deposited directly into a Federal Treasury account. Checks will be issued for refunds at the SMC. To track all financial data, information gathered on science users will be maintained in BAAS accounts. Receivable accounts report on science user accounts; and general ledger on summary assets and liabilities.

Mode Management: Mode Management addresses the execution of various system activities, whether they are functioning sequentially or simultaneously, so that the execution of one activity does not interfere with the execution of another. Examples of these activities, i.e. modes, include Operations, Testing, and Training. In addition, data integrity must be maintained between modes. The infrastructure that will ensure data and process separation and distinction between modes will be provided through the use of hierarchical name space registration, a process framework, and scripts. This is a cross-functional, cross-release effort that involves almost every CSCI. The Mode Management Service within MSS will:

5. Provide methods for initiating and terminating a given mode execution.
6. Enable the Fault Management Service and Performance Management Service to monitor each mode.
7. Enable the operator to control startup/shutdown/suspend/resume activities for each monitorable entity by utilizing the CSS provided life cycle services.

5.11.2 Management Subsystem Class Descriptions

5.11.2.1 Class EcAgEvent

Synopsis:

No Parent Class
 Distributed Object
 Is Associated With:
 Class: MsMdManager(Private) communicateswith
 Class: EcAgTuple(Private) contains
 Class: ECSApplication(Private) createdby
 Class: ECSApplication(Private) creates

Description:

The EcAgEvent defines a distributed object. It provides the capability to dispatch events for orderly and prompt resolution should events occur. The SNMP protocol provides the capability to send traps from agent to

SNMP manager. But, the traps are not secure and not reliable. The solution to these problems are using DCE RPC as the transport mechanism for security reasons and sending the traps from MSS Server to the management framework locally. The COTS HP OpenView guarantees the delivery of traps local on one host by using IPC as opposed to UDP. The ECS applications, the EcAgProxy agent, and the MsAgMonitor of the MsAgSubagent can send event notifications to the MsAgSubagent. The MsAgSubagent logs every event into MSS log file. Then, if the severity of the event equals to or is higher than the infoLevel variable, it sends this event notification further to the MsAgDeputy on the MSS Server which in turn convert the event to an SNMP trap and send it locally to the management framework.

Attributes:

`category`

Privilege: Private

Data Type: EcTAgEventCategory

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the category of the event.

`csci`

Privilege: Private

Data Type: EcTAgEventCsci

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the division within subsystem. i.e. Identifies the CSCI where the event occurred.

`nAppID`

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the application ID. i.e. The identifier of the application package

`nInstanceID`

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the instanceID. It is either the application's or the program's Instance ID.

nMsgNum

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the message number (related to message). i.e. Information, Warning, Error, Fatal, Critical

nNumTuples

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the number of tuples in this event.

nProcID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the process ID. i.e. ID of the process which sends out the event

nProgID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the program ID. i.e. The identifier of the program

nType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This method represents the event type. i.e. System error, Startup, Stop, Process failed, Threshold exceeded, Access attempts

rsEventID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the event ID, set by EcAgEvent constructor using UUID generator. i.e.The UUID of the event

rsMgmtSvrObjID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the object ID, server object has a unique identifier, UUID (filled in by EcAgManager) when EcAgManager is created by DCE. i.e.The object ID of the EcAgManager of the application.

rsMode

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the event mode, i.e. test mode or operational mode. Its string length is at most 8 characters.

rsMsg

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the message. i.e.The message of the event

rsTransactionID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the transaction ID. i.e.UUID of the transaction

rsTransactionParentID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the transaction's parent ID. i.e.UUID of the parent transaction

rtTimestamp

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the timestamp of the event.

severity

Privilege: Private

Data Type: EcTagEventSeverity

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the severity level of the event.

subsys

Privilege: Private

Data Type: EcTagEventSubsys

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the subsystem for the event.

tupleList

Privilege: Private

Data Type: RWSlistCollectables

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the tuple which makes up the event.

Operations:

void EcAgEvent ()

Privilege: Public

No Inheritance

The default constructor is only used by RWDECLARE. This method is required by Rogue Wave.

```
void EcAgEvent (Cat,Typ,Sev,Csci  
opt:MgNm,Mg,SSys,App,Prog,Inst,Proc,Trans, TransP)
```

Privilege: Public

No Inheritance

This method represents the constructor. It uses the values in the argument list to set the corresponding attributes.

```
void EcAgEvent (const EcAgEvent &event)
```

Privilege: Public

No Inheritance

This method represents the copy constructor.

```
EcTVoid Flatten (EcTInt nBufLen, EcTUChar *szBuf)
```

Privilege: Public

No Inheritance

This operation flattens the event.

```
EcTInt GetAppID ()
```

Privilege: Public

No Inheritance

This method gets the application ID.

```
EcTAgEventCategory GetCategory ()
```

Privilege: Public

No Inheritance

This method gets the event category.

```
EcTInt GetCsci ()
```

Privilege: Public

No Inheritance

This method gets the division within subsystem.

```
RWCString& GetEventID ()
```

Privilege: Public

No Inheritance

This method gets the event ID, set by Event Manager using UUID generator.

`EcTInt GetInstanceID ()`

Privilege: Public

No Inheritance

This method gets the instanceID is either the application's or the program's Instance ID.

`RWCString& GetMgmtSvrObjID ()`

Privilege: Public

No Inheritance

This method gets the object ID, server object has a unique identifier, UUID (filled in by Manager) when EcAgManager is created by DCE.

`RWCString GetMode ()`

Privilege: Public

No Inheritance

This method gets the event mode, i.e. test mode or operational mode.

`RWCString& GetMsg ()`

Privilege: Public

No Inheritance

This method gets the message.

`EcTInt GetMsgNum ()`

Privilege: Public

No Inheritance

This method gets the message number (related to message).

`EcTInt GetNumTuples ()`

Privilege: Public

No Inheritance

This method gets the number of tuples in linked list.

`EcTInt GetProcID ()`

Privilege: Public

No Inheritance

This method gets the process ID.

`EcTInt GetProgID ()`

Privilege: Public

No Inheritance

This method gets the program ID.

EcTagEventSeverity GetSeverity ()

Privilege: Public

No Inheritance

This operation gets the severity level of the event.

EcTagEventSubsys GetSubsys ()

Privilege: Public

No Inheritance

This method gets the subsystem of person creating event.

RWTime& GetTimestamp ()

Privilege: Public

No Inheritance

This method gets the event timestamp.

RWCString & GetTransactionID ()

Privilege: Public

No Inheritance

This method gets the transaction ID.

RWCString& GetTransactionParentID ()

Privilege: Public

No Inheritance

This method gets the transaction's parent ID.

EcAgTuple& GetTuple ()

Privilege: Public

No Inheritance

This method gets a tuple into the linked list of tuples. It will call Rogue Wave's 'get' function to retrieve an element from the rogue wave linked list.

RWSlistCollectables& GetTupleList ()

Privilege: Public

No Inheritance

This method gets a list of tuples.

EcTInt GetType ()

Privilege: Public

No Inheritance

This method gets the event type.

`EctVoid InsertTuple (EcAgTuple &tuple)`

Privilege: Public

No Inheritance

This operation inserts a tuple into an event

`EctVoid Restore (EctInt nBufLen, EctUChar *szBuf)`

Privilege: Public

No Inheritance

This operation restores from a flattened state.

`EctVoid SetAppID (EctInt nNewAppID)`

Privilege: Public

No Inheritance

This method sets the application ID.

`EctInt SetCategory (EcTagEventCategory newCategory)`

Privilege: Public

No Inheritance

This operation sets the category of the event.

`EctVoid SetCsci (EctInt nNewCsci)`

Privilege: Public

No Inheritance

This method sets the division within subsystem.

`EctVoid SetEventID ()`

Privilege: Public

No Inheritance

This method sets the event ID, set by Event Manager using UUID generator.

`EctVoid SetInstanceID (EctInt nNewInstanceID)`

Privilege: Public

No Inheritance

This method sets the instance ID.

`EctVoid SetMgmtSvrObjID (RWCString &rsNewMgmtSvrObjID)`

Privilege: Public

No Inheritance

This operation sets the management server object ID.

`EcUtStatus SetMode (RWCString rsNewMode)`

Privilege: Public

No Inheritance

This operation sets the mode of the event.

`EcTVoid SetMsg (RWCString &rsNewMsg)`

Privilege: Public

No Inheritance

This operation sets the event message string.

`EcTVoid SetMsgNum (EcTInt nNewMsgNum)`

Privilege: Public

No Inheritance

This method sets the message number (related to message).

`EcTVoid SetProcID (EcTInt nNewProcID)`

Privilege: Public

No Inheritance

This method sets the process ID.

`EcTVoid SetProgID (EcTInt nNewProgID)`

Privilege: Public

No Inheritance

This method sets the program ID.

`EcTInt SetSeverity (EcTagEventSeverity newSeverity)`

Privilege: Public

No Inheritance

This operation sets the severity level of the event.

`EcTInt SetSubsys (EcTagEventSubsys newSubsys)`

Privilege: Public

No Inheritance

this operation sets the subsystem of the event.

`EcTVoid SetTimestamp ()`

Privilege: Public

No Inheritance

This method sets the event timestamp.

`EcTVoid SetTimestamp (RWTime rtNewTime)`

Privilege: Protected Operation

No Inheritance

This operation sets the event timestamp to a new time.

`EcTVoid SetTransactionID (RWCString &rsNewTransactionID)`

Privilege: Public

No Inheritance

This operation sets the transaction ID of the event.

`EcTVoid SetTransactionParentID (RWCString
&rsNewTransactionParentID)`

Privilege: Public

No Inheritance

This operation sets the transaction ID for the parent transaction of the event.

`EcTVoid SetType (EcTInt nNewType)`

Privilege: Public

No Inheritance

This method sets the event type.

`RWSpace binaryStoreSize ()`

Privilege: Public

No Inheritance

This method returns the number of bytes used by the virtual function `saveGuts (RWFile&)` to store an object.

`friend ostream& operator << (ostream& stream, EcAgEvent&
event)`

Privilege: Public

No Inheritance

`friend istream& operator >> (istream& stream, EcAgEvent&
event)`

Privilege: Public

No Inheritance

`EcAgEvent& operator= (const EcAgEvent& event)`

Privilege: Public

No Inheritance

This method copies an event.

`EcTVoid restoreGuts (RWFile& f)`

Privilege: Public

No Inheritance

This method reads an object's state from an binary file, using RWFile, replacing the previous state.

`EcTVoid restoreGuts (RWvistream& stream)`

Privilege: Public

No Inheritance

This method reads an object's state from an input stream, replacing the previous state.

`EcTVoid saveGuts (RWFile& f)`

Privilege: Public

No Inheritance

This method writes an object's state to a binary file, using class RWFile.

`EcTVoid saveGuts (RWvostream& stream)`

Privilege: Public

No Inheritance

This method writes an object's state to an output stream. virtual

`EcTVoid saveGuts (RWvostream&) const {};`

`void ~EcAgEvent ()`

Privilege: Public

No Inheritance

This method specifies the destructor of the class.

5.11.2.2 Class EcAgProxy

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

Class: COTS(Private) manages

Class: EcAgCotsLog(Private) monitors

Class: EcAgCOTSMgrFactory(Private) usedby

Class: EcAgCOTSMgrFactory(Private) usedby

Class: EcAgManager(Private) usedby

Class: EcAgCOTSMgrFactory(Private) uses

Class: EcAgCOTSMgrFactory(Private) uses

Class: EcAgManager(Private) uses

Class: MsAgSubAgent(Public) uses

Description:

This object class is primarily for COTS' manageability. It includes the MSS instrumentation class library to enable the manageability of the COTS product. The front-end of this object is the MSS instrumentation code. The back-end of it is the interface to the COTS. It is unique to every COTS. In security management, the logs of COTS are monitored by this object. If an security event occurs, this object has to detect the incident and send out an event notification to the MsAgSubagent.

Attributes:

modeB

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute contains the mode in which the application is executing under. It identifies functional activity(operational, testing, training).

Lock

Privilege: Private

Data Type: DCEPthreadMutex

Default Value: NOT IDENTIFIED

No Inheritance

lock for the proxyEntry ordered

entryVector

Privilege: Private

Data Type: RWOrdered

Default Value: NOT IDENTIFIED

No Inheritance

RW Ordered of proxy entries

pCOTSMgrFactory

Privilege: Private

Data Type: EcAgCOTSMgrFactory*

Default Value: NOT IDENTIFIED

No Inheritance

A pointer to the COTSMgrFactory object.

rsName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Name of the proxy agent.

Operations:

`RWCString CreateCDSName ()`

Privilege: Private

No Inheritance

Creates a CDS the object uses to register into the CDS.

`void EcAgProxy (RWCString rsNewName)`

Privilege: Public

No Inheritance

This operation represents the constructor of the MsAgProxy class.

`EcUtStatus Listen ()`

Privilege: Public

No Inheritance

Listens for incoming requests to be monitored.

`EcTVoid Ping ()`

Privilege: Public

No Inheritance

listens for incoming requests to be monitored

`EcTVoid ResumeManager (EcTUChar* szProxyName, EcTLongInt
nNewAppID, EcTLongInt nNewProgID, EcTLongInt nNewProcID,
EcTLongInt nNewInstID)`

Privilege: Public

No Inheritance

This distributed method is executed by ProxyComm. It resumes the manager specified by the ProcID passed in.

`EcTVoid SaveSuspendedEntryVector ()`

Privilege: Private

No Inheritance

This function saves the suspended entry to a file so that it can be read back in at resume time.

`EcTVoid SetManagerFactory (EcAgCOTSMgrFactory
*pNewCOTSMgrFactory)`

Privilege: Public

No Inheritance

Sets the default manager factory object.

```
static pthread_adr_t SpawnManagerProcess (pthread_addr_t  
pData)
```

Privilege: Private

No Inheritance

executes a single manager passed in a MsAgProxyParam which is in turn, passwd in as a EcTVoid*. This method is executed by a newly spawned thread

```
EcTVoid StartManager (EcTUChar* szProxyName, EcTLongInt  
nNewAppID, EcTLongInt nNewProgID, EcTLongInt nNewProcID,  
EcTLongInt nNewInstID)
```

Privilege: Private

No Inheritance

rpc method to request a COTS process to be monitored

```
EcTVoid StopManager (EcTUChar* szProxyName, EcTLongInt  
nNewAppID, EcTLongInt nNewProgID, EcTLongInt nNewProcID,  
EcTLongInt nNewInstID)
```

Privilege: Public

No Inheritance

rpc method to request a COTS process to be monitored

```
EcTVoid SuspendManager (EcTUChar* szProxyName, EcTLongInt  
nNewAppID, EcTLongInt nNewProgID, EcTLongInt nNewProcID,  
EcTLongInt nNewInstID)
```

Privilege: Public

No Inheritance

This distributed method is executed by ProxyComm. It suspends the manager specified by the ProcID passed in.

5.11.2.3 Class EcDAAC

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: MsFlManager(Private) ismanagedby

Description:

This public class provides methods to respond to requests from other subsystems for the status of resources (processors and their associated disks).

Attributes:

Operations:

```
void EcDAAC ()
```

Privilege: Public

No Inheritance

This method represents the constructor of the class.

```
void GetAppList (char* AppName)
```

Privilege: Public

No Inheritance

This method returns a list of application classes (dependent) given an application class name as input.

```
void Get_CPU_List (char* Filter)
```

Privilege: Public

No Inheritance

This method takes a filter as an argument, and returns a list of hosts matching the filter criteria.

```
void Get_Disk_List (char* CPU_ID)
```

Privilege: Public

No Inheritance

This method returns a list of disks attached to a specified processor. The processor is specified by the argument CPU_ID.

```
void ~EcDAAC ()
```

Privilege: Public

No Inheritance

This method represents the destructor of the class.

5.11.2.4 Class EcExtSysIFB

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: MsScManager(Private) exchangenotificationswith

Description:

This class represents the interface to external systems such as NSI.

Attributes:

Operations:

```
void EcExtSysIFB ()
```

Privilege: Public

No Inheritance

This is the default constructor for this class.

```
void Listen ()
```

Privilege: Public

No Inheritance

This method listens for an SNMP trap or a TCP socket, from an external subsystem.

```
void Send (char* destination)
```

Privilege: Public

No Inheritance

This method sends a mail message to the external system as specified by the destination field.

```
void ~EcExtSysIFB ()
```

Privilege: Public

No Inheritance

This is the destructor for this class.

5.11.2.5 Class EcOrder

Synopsis:

Parent Class: EcRequest

Is Not A Distributed Object

Is Associated With:

Class: EcOrderEvent(Public) generates

Class: EcPfManagedServer(Public) processnon-statechangeorderevents

Description:

This is a public class which is used by ECS applications to collect resource utilizations associated with an order type of request. The class is also used by the application to report the state of the order when the order state changes. Objects in this class should remain until the application has finished processing the associated order request. An order type of request is the top most root of a hierarchy of sub-classes and services that is associated with a Product Data Order request from an ECS user.

Attributes:

distList

Privilege: Private

Data Type: DistListType

Default Value: NOT IDENTIFIED

No Inheritance

The distribution list for the product.

estimatedPrice

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The price which was reported to the ECS user and the price which is to be decremented from the available balance of the user.

homeDAAC

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The site at which the user is registered, who placed the product order.

orderUR

Privilege: Private

Data Type: EcTUR

Default Value: NOT IDENTIFIED

No Inheritance

This is the UR for the order which is reported back to the ECS user. This is stored in the tracking database so that the order tracking information can be retrieved by the Order UR.

shipAddress

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Mailing address that the products produced/retrieved for the order are to be shipped.

shipMethod

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The method of shipment - how the product(s) are to be sent to the requesting user.

shipToName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The name to which the products are to be addressed.

userId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The unique ECS user identification of the user who placed the order.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the cpu counter of the system when resource utilization collection is started. This value is then used to calculate the amount of cpu which was used during the collection period.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of disk utilization which has been used while processing this request.

idleTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been idle.

ioUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the I/O utilization counter of the system when resource utilization collection is started. This value is then used to calculate the amount of I/O utilization which was used during the collection period.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of I/O utilization which has been used while processing this request.

lastEventID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the event identification of the last event that was reported to the MSS event logging capability. This event ID allows an operator to browse through the event log chain for the request in order to show the history of state changes as well as to see any other significant events associated with this request.

requestDate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the time at which the current state was changed.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The type of the request being processed.

Operations:

```
void EcOrder (RWCString description, enum type, enum state,
EcTUR orderUR, RWCString userId, RWCString homeDAAC,
RWCString shipAddress, RWCString shipToName, RWCString
shipMethod, DiskListType distList, EcTLong estimatedPrice)
```

Privilege: Public

No Inheritance

The constructor for the class. The order related information is initialized and the information, including the starting state, are send to the request tracking server. In addition, the resource utilization counters are initialized to zero.

```
void ~EcOrder ()
```

Privilege: Public

No Inheritance

Default destructor for the class. The collected resource utilization and the final state of the order are sent to the request tracking server.

```
void EcRequest (RWCString description, enum type, enum state)
```

Privilege: Public

Inherited From: EcRequest

This is the constructor for the object. This method sets the attributes of the request object to the passed values and initializes the resource utilization totals to zero.

EcTLong GetDiskUtilization ()

Privilege: Public

Inherited From: EcRequest

Returns the current value of the disk utilization attribute.

EcTLong GetRequestID ()

Privilege: Public

Inherited From: EcRequest

Returns the value of the request ID attribute.

trackingStateType GetState ()

Privilege: Public

Inherited From: EcRequest

Returns the current value of the request state attribute.

EcTVoid SetDiskUtilization (diskUtilization)

Privilege: Public

Inherited From: EcRequest

EcTVoid SetState (state)

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

EcTVoid SetState (trackingStateType newState)

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

EcTVoid StartCollecting ()

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and stores them in attributes. This method should be called at the beginning of a method which processes the request associated with this object.

`EcTvoid StopCollecting ()`

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and subtracts from them the associated values stored in the attributes. The resulting value will be added to the running total utilization attribute. This method should be called at the end of a method which processes the request associated with this object.

`void ~EcRequest ()`

Privilege: Public

Inherited From: EcRequest

This is the default destructor of the object. This method cleans up any memory which was allocated to attributes within this object.

`ProcessEvent (RWCString eventDescription)`

Privilege: Protection Not Identified

Inherited From: EcRequest

No description

5.11.2.6 Class EcOrderEvent

Synopsis:

Parent Class: EcRequestEvent

Distributed Object

Is Associated With:

Class: EcOrder(Public) generates

Class: MsAcTrackingMgr(Private) updatetrackinginfo

Description:

This is a public, distributed object whose purpose is to report information collected about an order type of request. An order type of request is the root of a request hierarchy structure that was generated based on a Product Order request from an ECS user. Objects of this class are created with the information to be reported and processed (sent to the request tracking server) and then destroyed. These objects only need to stay around long enough for the event to be processed.

Attributes:

`distList`

Privilege: Private

Data Type: DistListType

Default Value: NOT IDENTIFIED

No Inheritance

The distribution list for the product.

estimatedPrice

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The price which was reported to the ECS user and the price which is to be decremented from the available balance of the user.

homeDAAC

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The site at which the user is registered, who placed the product order.

orderUR

Privilege: Private

Data Type: EcTUR

Default Value: NOT IDENTIFIED

No Inheritance

This is the UR for the order which is reported back to the ECS user. This is stored in the tracking database so that the order tracking information can be retrieved by the Order UR.

shipAddress

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Mailing address that the products produced/retrieved for the order are to be shipped.

shipMethod

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The method of shipment - how the product(s) are to be sent to the requesting user.

shipToName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The name to which the products are to be addressed.

userId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The unique ECS user identification of the user who placed the order.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of disk utilization which has been used while processing this request.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of I/O utilization which has been used while processing this request.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The date/time at which the request started to be processed.

sleepTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has not been actively processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the time at which the current state was changed.

totalTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the total amount of real-time which was required to process the request.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The type of the request being processed.

category

Privilege: Private

Data Type: EcTAgeventCategory

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the category of the event.

csci

Privilege: Private

Data Type: EcTAgeventCsci

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the division within subsystem. i.e. Identifies the CSCI where the event occurred.

nAppID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the application ID. i.e. The identifier of the application package

nInstanceID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the instanceID. It is either the application's or the program's Instance ID.

nMsgNum

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message number (related to message). i.e. Information, Warning, Error, Fatal, Critical

nNumTuples

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the number of tuples in this event.

nProcID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the process ID. i.e. ID of the process which sends out the event

nProgID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the program ID. i.e. The identifier of the program

nType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This method represents the event type. i.e. System error, Startup, Stop, Process failed, Threshold exceeded, Access attempts

rsEventID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event ID, set by EcAgEvent constructor using UUID generator. i.e.The UUID of the event

rsMgmtSvrObjID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the object ID, server object has a unique identifier, UUID (filled in by EcAgManager) when EcAgManager is created by DCE. i.e.The object ID of the EcAgManager of the application.

rsMode

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event mode, i.e. test mode or operational mode. Its string length is at most 8 characters.

rsMsg

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message. i.e.The message of the event

rsTransactionID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction ID. i.e.UUID of the transaction

rsTransactionParentID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction's parent ID. i.e.UUID of the parent transaction

rtTimestamp

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the timestamp of the event.

severity

Privilege: Private

Data Type: EcTagEventSeverity

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the severity level of the event.

subsys

Privilege: Private

Data Type: EcTagEventSubsys

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the subsystem for the event.

tupleList

Privilege: Private

Data Type: RWSlistCollectables

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the tuple which makes up the event.

Operations:

```
EcOrderEvent (EcTLong itemID, RWString description,  
requestType type, requestStateType state, EcTTime  
timeStateUpdated, requestCost rCost, EcTUR orderUR, RWString  
userId, RWString homeDAAC, distList, EcTLong estimatedPrice)
```

Privilege: Protection Not Identified

No Inheritance

Constructor for the class. The information to be reported to the request tracking server is initialized.

```
EcTVoid ProcessOrderEvent ()
```

Privilege: Public

No Inheritance

This method takes the information which was set in the class and sends the information to the request tracking server.

```
void ~EcOrderEvent ()
```

Privilege: Public

No Inheritance

Default constructor for the class. This method cleans up the order event object by removing any memory allocated for the attributes.

```
EcRequest (EcTLong, requestID, RWCString descrip, enum type,  
EcTTime timeOfLast, EcTLong cpuUtil, EcTLong ioUtil, EcTLong  
diskUtil, EcTTime rqStart, EcTTime SleepT, EcTTime activeT,  
EcTTime totalTime)
```

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

```
~EcRequest ()
```

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

```
void EcAgEvent ()
```

Privilege: Public

Inherited From: EcAgEvent

The default constructor is only used by RWDECLARE. This method is required by Rogue Wave.


```
void EcAgEvent (Cat,Typ,Sev,Csci  
opt:MgNm,Mg,SSys,App,Prog,Inst,Proc,Trans, TransP)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the constructor. It uses the values in the argument list to set the corresponding attributes.

```
void EcAgEvent (const EcAgEvent &event)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the copy constructor.

```
EcTVoid Flatten (EcTInt nBufLen, EcTUChar *szBuf)
```

Privilege: Public

Inherited From: EcAgEvent

This operation flattens the event.

```
EcTInt GetAppID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the application ID.

```
EcTAgEventCategory GetCategory ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event category.

```
EcTInt GetCsci ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the division within subsystem.

```
RWCString& GetEventID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event ID, set by Event Manager using UUID generator.

`EcTInt GetInstanceID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the instanceID is either the application's or the program's Instance ID.

`RWCString& GetMgmtSvrObjID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the object ID, server object has a unique identifier, UUID (filled in by Manager) when EcAgManager is created by DCE.

`RWCString GetMode ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the event mode, i.e. test mode or operational mode.

`RWCString& GetMsg ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the message.

`EcTInt GetMsgNum ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the message number (related to message).

`EcTInt GetNumTuples ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the number of tuples in linked list.

`EcTInt GetProcID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the process ID.

`EcTInt GetProgID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the program ID.

EcTagEventSeverity GetSeverity ()

Privilege: Public

Inherited From: EcAgEvent

This operation gets the severity level of the event.

EcTagEventSubsys GetSubsys ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the subsystem of person creating event.

RWTime& GetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event timestamp.

RWCString & GetTransactionID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction ID.

RWCString& GetTransactionParentID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction's parent ID.

EcAgTuple& GetTuple ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a tuple into the linked list of tuples. It will call Rogue Wave's 'get' function to retrieve an element from the rogue wave linked list.

RWSlistCollectables& GetTupleList ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a list of tuples.

EcTInt GetType ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event type.

`EcTVoid InsertTuple (EcAgTuple &tuple)`

Privilege: Public

Inherited From: EcAgEvent

This operation inserts a tuple into an event

`EcTVoid Restore (EcTInt nBufLen, EcTUChar *szBuf)`

Privilege: Public

Inherited From: EcAgEvent

This operation restores from a flattened state.

`EcTVoid SetAppID (EcTInt nNewAppID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the application ID.

`EcTInt SetCategory (EcTagEventCategory newCategory)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the category of the event.

`EcTVoid SetCsci (EcTInt nNewCsci)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the division within subsystem.

`EcTVoid SetEventID ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event ID, set by Event Manager using UUID generator.

`EcTVoid SetInstanceID (EcTInt nNewInstanceID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the instance ID.

`EcTVoid SetMgmtSvrObjID (RWCString &rsNewMgmtSvrObjID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the management server object ID.

EcUtStatus SetMode (RWCString rsNewMode)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the mode of the event.

EcTVoid SetMsg (RWCString &rsNewMsg)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the event message string.

EcTVoid SetMsgNum (EcTInt nNewMsgNum)

Privilege: Public

Inherited From: EcAgEvent

This method sets the message number (related to message).

EcTVoid SetProcID (EcTInt nNewProcID)

Privilege: Public

Inherited From: EcAgEvent

This method sets the process ID.

EcTVoid SetProgID (EcTInt nNewProgID)

Privilege: Public

Inherited From: EcAgEvent

This method sets the program ID.

EcTInt SetSeverity (EcTAgEventSeverity newSeverity)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the severity level of the event.

EcTInt SetSubsys (EcTAgEventSubsys newSubsys)

Privilege: Public

Inherited From: EcAgEvent

this operation sets the subsystem of the event.

EcTVoid SetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method sets the event timestamp.

`EcTVoid SetTimestamp (RWTime rtNewTime)`

Privilege: Protected Operation

Inherited From: EcAgEvent

This operation sets the event timestamp to a new time.

`EcTVoid SetTransactionID (RWCString &rsNewTransactionID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID of the event.

`EcTVoid SetTransactionParentID (RWCString
&rsNewTransactionParentID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID for the parent transaction of the event.

`EcTVoid SetType (EcTInt nNewType)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event type.

`RWSpace binaryStoreSize ()`

Privilege: Public

Inherited From: EcAgEvent

This method returns the number of bytes used by the virtual function `saveGuts (RWFile&)` to store an object.

`friend ostream& operator << (ostream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`friend istream& operator >> (istream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`EcAgEvent& operator= (const EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

This method copies an event.

`EcTVoid restoreGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an binary file, using RWFile, replacing the previous state.

`EcTVoid restoreGuts (RWvistream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an input stream, replacing the previous state.

`EcTVoid saveGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to a binary file, using class RWFile.

`EcTVoid saveGuts (RWvostream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to an output stream. virtual
`EcTVoid saveGuts (RWvostream&) const {};`

`void ~EcAgEvent ()`

Privilege: Public

Inherited From: EcAgEvent

This method specifies the destructor of the class.

5.11.2.7 Class EcPfManagedServer

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: EcOrder(Public) processnon-statechangeorderevents

Class: EcService(Public) processnon-statechangeserviceevents

Class: EcSubOrder(Public) processnon-statechangesub-orderevents

Description:

This is the container class that starts up the event Manager, table Manager, monitor, port monitor, discoverer, subagent configuration, static buffer, and the deputy gate. This class also starts a thread that triggers scheduled events (i.e. polling ECS application's performance metrics).

Attributes:

Operations:

5.11.2.8 Class EcPriceTableB

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
Class: MsAcTrackingMgr(Private) providepriceforcancelledrequests

Description:

This class represents a public and distributed class that holds the prices of every billable item in the ECS inventory of products and services. Price of hard media and standard shipping costs are also maintained in this table.

Attributes:

my4mmTapePrice

Privilege: Private
Data Type: EcCurrency
Default Value: NOT IDENTIFIED
No Inheritance

This attribute represents the price charged for an 4mm tape that is part of an order for an ECS data product request.

my8mmTapePrice

Privilege: Private
Data Type: EcCurrency
Default Value: NOT IDENTIFIED
No Inheritance

This attribute represents the price charged for an 8mm tape that is part of an order for an ECS data product request.

myCDPrice

Privilege: Private
Data Type: EcCurrency
Default Value: NOT IDENTIFIED
No Inheritance

This attribute represents the price charged for one Compact Disc (CD) that will be used to store data that is part of an order for an ECS data product request.

myExprsShipPrice

Privilege: Private

Data Type: EcCurrency

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the price charged for shipping an ECS data product request by the express mail method contained in the user's profile information.

myProductdPrice

Privilege: Private

Data Type: EcCurrency

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the price charged for a chargeable and identifiable ECS data product. The format and content of existing DAACs pricing lists of products, media and services will be incorporated into the structure of the EcPriceTableB as much as possible. Identifying products by a granule Id, size of granule and other price related factors will also be considered by the ECS system with the actual price guidelines for such attributes determined by an EOSDIS Pricing Policy committee.

myServicePrice

Privilege: Private

Data Type: EcCurrency

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the price charged for a service (such as dataset subsetting) that is required in the process of fulfilling an order for an ECS data product request.

myStdrrdShipPrice

Privilege: Private

Data Type: EcCurrency

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the price charged for shipping an ECS data product request by the normal mail method contained in the user's profile information.

Operations:

`EcPriceTableB ()`

Privilege: Protection Not Identified

No Inheritance

`void ProvidePrices (RWCString userID; RWCString requestID,
GlParameterList)`

Privilege: Public

No Inheritance

This method represents the summation of all the parameters in a user's request for a data product request including shipping charges. The global parameter list, `GlParameterList` passed in will contain the price element IDs corresponding to the type of product, type of service(s) required to satisfy the request, the type and number of media, and shipping method. This method will reference standard price entries for each of these parameters and arrive at a total price for the given request.

`~EcPriceTableB ()`

Privilege: Protection Not Identified

No Inheritance

5.11.2.9 Class `EcRequest`

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

None

Description:

This is an abstract class which represents all types of requests which are tracked in the ECS system. This class contains the attributes and operations which are common to all of the request types. The objects which are created from the sub-classes are used to track resource utilization of the associated system request types as well as to maintain and report the current state of the associated system request to the request tracking server.

Attributes:

`activeTime`

Privilege: Private

Data Type: `EcTTime`

Default Value: NOT IDENTIFIED

No Inheritance

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is set to the current value of the cpu counter of the system when resource utilization collection is started. This value is then used to calculate the amount of cpu which was used during the collection period.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The running total amount of disk utilization which has been used while processing this request.

idleTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is a resource utilization counter which contains the approximate amount of real-time that the request has been idle.

ioUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is set to the current value of the I/O utilization counter of the system when resource utilization collection is started. This value is then used to calculate the amount of I/O utilization which was used during the collection period.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The running total amount of I/O utilization which has been used while processing this request.

lastEventID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This is the event identification of the last event that was reported to the MSS event logging capability. This event ID allows an operator to browse through the event log chain for the request in order to show the history of state changes as well as to see any other significant events associated with this request.

requestDate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

The date/time at which the request started to be processed.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

The date/time at which the request started to be processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is the time at which the current state was changed.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

The type of the request being processed.

Operations:

void EcRequest (RWCString description, enum type, enum state)

Privilege: Public

No Inheritance

This is the constructor for the object. This method sets the attributes of the request object to the passed values and initializes the resource utilization totals to zero.

EcTLong GetDiskUtilization ()

Privilege: Public

No Inheritance

Returns the current value of the disk utilization attribute.

`EctLong GetRequestID ()`

Privilege: Public

No Inheritance

Returns the value of the request ID attribute.

`trackingStateType GetState ()`

Privilege: Public

No Inheritance

Returns the current value of the request state attribute.

`EctVoid SetDiskUtilization (diskUtilization)`

Privilege: Public

No Inheritance

`EctVoid SetState (state)`

Privilege: Public

No Inheritance

This method sets the current value of the request state attribute to the passed value.

`EctVoid SetState (trackingStateType newState)`

Privilege: Public

No Inheritance

This method sets the current value of the request state attribute to the passed value.

`EctVoid StartCollecting ()`

Privilege: Public

No Inheritance

This method reads from the system the current values of particular resource counters and stores them in attributes. This method should be called at the beginning of a method which processes the request associated with this object.

`EctVoid StopCollecting ()`

Privilege: Public

No Inheritance

This method reads from the system the current values of particular resource counters and subtracts from them the associated values stored in the attributes. The resulting value will be added to the running total utilization attribute. This method should be called at the end of a method which processes the request associated with this object.

`void ~EcRequest ()`

Privilege: Public

No Inheritance

This is the default destructor of the object. This method cleans up any memory which was allocated to attributes within this object.

`ProcessEvent (RWCString eventDescription)`

Privilege: Protection Not Identified

No Inheritance

No description

5.11.2.10 Class EcRequestEvent

Synopsis:

Parent Class: EcAgEvent

Distributed Object

Is Associated With:

This class is derived from the class EcAgEvent

Description:

This is an abstract class which represents all types of requests event objects which are used to report information collected about a request. This class contains the attributes and operations which are common to all of the request event types. The objects which are created from the sub-classes are used to report resource utilization of the associated system request types as well as to report the current state of the associated system request to the request tracking server.

Attributes:

`activeTime`

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

`cpuUtilization`

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
A textual description of the request.

diskUtilization

Privilege: Private
Data Type: EcTLong
Default Value: NOT IDENTIFIED
No Inheritance
The running total amount of disk utilization which has been used while processing this request.

ioUtilization

Privilege: Private
Data Type: EcTLong
Default Value: NOT IDENTIFIED
No Inheritance
The running total amount of I/O utilization which has been used while processing this request.

requestID

Privilege: Private
Data Type: EcTLong
Default Value: NOT IDENTIFIED
No Inheritance
A unique identification of the request.

requestStartTime

Privilege: Private
Data Type: EcTTime
Default Value: NOT IDENTIFIED
No Inheritance
The date/time at which the request started to be processed.

sleepTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is a resource utilization counter which contains the approximate amount of real-time that the request has not been actively processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is the time at which the current state was changed.

totalTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is the total amount of real-time which was required to process the request.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

The type of the request being processed.

category

Privilege: Private

Data Type: EcTAgEventCategory

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the category of the event.

csci

Privilege: Private

Data Type: EcTAgEventCsci

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the division within subsystem. i.e. Identifies the CSCI where the event occurred.

nAppID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the application ID. i.e. The identifier of the application package

nInstanceID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the instanceID. It is either the application's or the program's Instance ID.

nMsgNum

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message number (related to message). i.e. Information, Warning, Error, Fatal, Critical

nNumTuples

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the number of tuples in this event.

nProcID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the process ID. i.e. ID of the process which sends out the event

nProgID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the program ID. i.e. The identifier of the program

nType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This method represents the event type. i.e. System error, Startup, Stop, Process failed, Threshold exceeded, Access attempts

rsEventID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event ID, set by EcAgEvent constructor using UUID generator. i.e. The UUID of the event

rsMgmtSvrObjID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the object ID, server object has a unique identifier, UUID (filled in by EcAgManager) when EcAgManager is created by DCE. i.e. The object ID of the EcAgManager of the application.

rsMode

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event mode, i.e. test mode or operational mode. Its string length is at most 8 characters.

rsMsg

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message. i.e. The message of the event

rsTransactionID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction ID. i.e. UUID of the transaction

rsTransactionParentID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction's parent ID. i.e. UUID of the parent transaction

rtTimestamp

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the timestamp of the event.

severity

Privilege: Private

Data Type: EcTagEventSeverity

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the severity level of the event.

subsys

Privilege: Private

Data Type: EcTagEventSubsys

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the subsystem for the event.

tupleList

Privilege: Private

Data Type: RWSlistCollectables

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the tuple which makes up the event.

Operations:

EcRequest (EcTLong, requestID, RWCString descrip, enum type, EcTTime timeOfLast, EcTLong cpuUtil, EcTLong ioUtil, EcTLong diskUtil, EcTTime rqStart, EcTTime SleepT, EcTTime activeT, EcTTime totalTime)

Privilege: Protection Not Identified

No Inheritance

No description

~EcRequest ()

Privilege: Protection Not Identified

No Inheritance

No description

void EcAgEvent ()

Privilege: Public

Inherited From: EcAgEvent

The default constructor is only used by RWDECLARE. This method is required by Rogue Wave.

```
void EcAgEvent (Cat,Typ,Sev,Csci  
opt:MgNm,Mg,SSys,App,Prog,Inst,Proc,Trans, TransP)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the constructor. It uses the values in the argument list to set the corresponding attributes.

```
void EcAgEvent (const EcAgEvent &event)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the copy constructor.

```
EcTVoid Flatten (EcTInt nBufLen, EcTUChar *szBuf)
```

Privilege: Public

Inherited From: EcAgEvent

This operation flattens the event.

```
EcTInt GetAppID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the application ID.

```
EcTAgEventCategory GetCategory ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event category.

```
EcTInt GetCsci ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the division within subsystem.

```
RWCString& GetEventID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event ID, set by Event Manager using UUID generator.

```
EcTInt GetInstanceID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the instanceID is either the application's or the program's Instance ID.

RWCString& GetMgmtSvrObjID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the object ID, server object has a unique identifier, UUID (filled in by Manager) when EcAgManager is created by DCE.

RWCString GetMode ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event mode, i.e. test mode or operational mode.

RWCString& GetMsg ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the message.

EcTInt GetMsgNum ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the message number (related to message).

EcTInt GetNumTuples ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the number of tuples in linked list.

EcTInt GetProcID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the process ID.

EcTInt GetProgID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the program ID.

EcTAgEventSeverity GetSeverity ()

Privilege: Public

Inherited From: EcAgEvent

This operation gets the severity level of the event.

EcTagEventSubsys GetSubsys ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the subsystem of person creating event.

RWTime& GetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event timestamp.

RWCString & GetTransactionID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction ID.

RWCString& GetTransactionParentID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction's parent ID.

EcAgTuple& GetTuple ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a tuple into the linked list of tuples. It will call Rogue Wave's 'get' function to retrieve an element from the rogue wave linked list.

RWSlistCollectables& GetTupleList ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a list of tuples.

EcTInt GetType ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event type.

EcTVoid InsertTuple (EcAgTuple &tuple)

Privilege: Public

Inherited From: EcAgEvent

This operation inserts a tuple into an event

`EcTVoid Restore (EcTInt nBufLen, EcTUChar *szBuf)`

Privilege: Public

Inherited From: EcAgEvent

This operation restores from a flattened state.

`EcTVoid SetAppID (EcTInt nNewAppID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the application ID.

`EcTInt SetCategory (EcTAgEventCategory newCategory)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the category of the event.

`EcTVoid SetCsci (EcTInt nNewCsci)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the division within subsystem.

`EcTVoid SetEventID ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event ID, set by Event Manager using UUID generator.

`EcTVoid SetInstanceID (EcTInt nNewInstanceID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the instance ID.

`EcTVoid SetMgmtSvrObjID (RWCString &rsNewMgmtSvrObjID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the management server object ID.

`EcUtStatus SetMode (RWCString rsNewMode)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the mode of the event.

`EctVoid SetMsg (RWCString &rsNewMsg)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the event message string.

`EctVoid SetMsgNum (EctInt nNewMsgNum)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the message number (related to message).

`EctVoid SetProcID (EctInt nNewProcID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the process ID.

`EctVoid SetProgID (EctInt nNewProgID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the program ID.

`EctInt SetSeverity (EcTagEventSeverity newSeverity)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the severity level of the event.

`EctInt SetSubsys (EcTagEventSubsys newSubsys)`

Privilege: Public

Inherited From: EcAgEvent

this operation sets the subsystem of the event.

`EctVoid SetTimestamp ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event timestamp.

`EctVoid SetTimestamp (RWTime rtNewTime)`

Privilege: Protected Operation

Inherited From: EcAgEvent

This operation sets the event timestamp to a new time.

`EcTVoid SetTransactionID (RWCString &rsNewTransactionID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID of the event.

`EcTVoid SetTransactionParentID (RWCString
&rsNewTransactionParentID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID for the parent transaction of the event.

`EcTVoid SetType (EcTInt nNewType)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event type.

`RWSpace binaryStoreSize ()`

Privilege: Public

Inherited From: EcAgEvent

This method returns the number of bytes used by the virtual function `saveGuts (RWFile&)` to store an object.

`friend ostream& operator << (ostream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`friend istream& operator >> (istream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`EcAgEvent& operator= (const EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

This method copies an event.

`EcTVoid restoreGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an binary file, using RWFile, replacing the previous state.

`EcTVoid restoreGuts (RWvistream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an input stream, replacing the previous state.

`EcTVoid saveGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to a binary file, using class RWFile.

`EcTVoid saveGuts (RWvostream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to an output stream. virtual
`EcTVoid saveGuts (RWvostream&) const {};`

`void ~EcAgEvent ()`

Privilege: Public

Inherited From: EcAgEvent

This method specifies the destructor of the class.

5.11.2.11 Class EcService

Synopsis:

Parent Class: EcRequest

Is Not A Distributed Object

Is Associated With:

Class: EcServiceEvent(Public) generates

Class: EcPfManagedServer(Public) processnon-
statechangeserviceevents

EcOrder (Aggregation)

EcSubOrder (Aggregation)

Description:

This is a public class which is used by ECS applications to collect resource utilizations associated with service type of request. The class is also used by the application to report the state of the service when the

service state changes. Objects in this class should remain until the application has finished processing the associated service request. A service type of request is a request which is not associated with retrieving a specific product, a service could be spawned from a sub-order type of request if the ECS application spawns a set of processing which does not result in product. a service type of request could be the root of a service request hierarchy for tracking ECS requests which are not Product Orders.

Attributes:

homeDAAC

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The site at which the user is registered, who placed the product order.

parentId

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This is the request ID of the service or sub-order type of request which spawned this service type of request.

serviceUR

Privilege: Private

Data Type: EcTUR

Default Value: NOT IDENTIFIED

No Inheritance

This is the UR for the service which is reported back to the ECS user. This is stored in the tracking database so that the service tracking information can be retrieved by the Service UR. This attribute is only used if the service is the root of the service hierarchy.

userId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The unique ECS user identification of the user who placed the service. This attribute is only used if the service is the root of the service hierarchy.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the cpu counter of the system when resource utilization collection is started. This value is then used to calculate the amount of cpu which was used during the collection period.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of disk utilization which has been used while processing this request.

idleTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been idle.

ioUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the I/O utilization counter of the system when resource utilization collection is started. This value is then used to calculate the amount of I/O utilization which was used during the collection period.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of I/O utilization which has been used while processing this request.

lastEventID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the event identification of the last event that was reported to the MSS event logging capability. This event ID allows an operator to browse through the event log chain for the request in order to show the history of state changes as well as to see any other significant events associated with this request.

requestDate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the time at which the current state was changed.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The type of the request being processed.

Operations:

```
void EcService (RWCString description, enum type, enum state,  
EcTLong parentID, RWCString userId, EcTUR serviceUR,  
RWCString homeDAAC)
```

Privilege: Public

No Inheritance

The constructor for the class. The service related information is initialized and the information, including the starting state, are send to the request tracking server. In addition, the resource utilization counters are initialized to zero.

`void ~EcService ()`

Privilege: Public

No Inheritance

Default destructor for the class. The collected resource utilization and the final state of the service are sent to the request tracking server.

`void EcRequest (RWCString description, enum type, enum state)`

Privilege: Public

Inherited From: EcRequest

This is the constructor for the object. This method sets the attributes of the request object to the passed values and initializes the resource utilization totals to zero.

`EcTLong GetDiskUtilization ()`

Privilege: Public

Inherited From: EcRequest

Returns the current value of the disk utilization attribute.

`EcTLong GetRequestID ()`

Privilege: Public

Inherited From: EcRequest

Returns the value of the request ID attribute.

`trackingStateType GetState ()`

Privilege: Public

Inherited From: EcRequest

Returns the current value of the request state attribute.

`EcTVoid SetDiskUtilization (diskUtilization)`

Privilege: Public

Inherited From: EcRequest

`EcTVoid SetState (state)`

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

`EcTVoid SetState (trackingStateType newState)`

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

`EcTVoid StartCollecting ()`

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and stores them in attributes. This method should be called at the beginning of a method which processes the request associated with this object.

`EcTVoid StopCollecting ()`

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and subtracts from them the associated values stored in the attributes. The resulting value will be added to the running total utilization attribute. This method should be called at the end of a method which processes the request associated with this object.

`void ~EcRequest ()`

Privilege: Public

Inherited From: EcRequest

This is the default destructor of the object. This method cleans up any memory which was allocated to attributes within this object.

`ProcessEvent (RWCString eventDescription)`

Privilege: Protection Not Identified

Inherited From: EcRequest

No description

5.11.2.12 Class EcServiceEvent

Synopsis:

Parent Class: EcRequestEvent

Distributed Object

Is Associated With:

Class: EcService(Public) generates

Class: MsAcTrackingMgr(Private) updatetrackinginfo

Description:

This is a public, distributed object whose purpose is to report inform

Attributes:

homeDAAC

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The site at which the user is registered, who placed the product order or service.

parentId

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This is the request ID of the service or sub-order type of request which spawned this service type of request.

serviceUR

Privilege: Private

Data Type: EcTUR

Default Value: NOT IDENTIFIED

No Inheritance

This is the UR for the service which is reported back to the ECS user. This is stored in the tracking database so that the service tracking information can be retrieved by the Service UR. This attribute is only used if the service is the root of the service hierarchy.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of disk utilization which has been used while processing this request.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of I/O utilization which has been used while processing this request.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The date/time at which the request started to be processed.

sleepTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has not been actively processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the time at which the current state was changed.

totalTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the total amount of real-time which was required to process the request.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The type of the request being processed.

category

Privilege: Private

Data Type: EcTAgEventCategory

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the category of the event.

csci

Privilege: Private

Data Type: EcTAgEventCsci

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the division within subsystem. i.e. Identifies the CSCI where the event occurred.

nAppID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the application ID. i.e. The identifier of the application package

nInstanceID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the instanceID. It is either the application's or the program's Instance ID.

nMsgNum

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message number (related to message). i.e. Information, Warning, Error, Fatal, Critical

nNumTuples

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the number of tuples in this event.

nProcID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the process ID. i.e. ID of the process which sends out the event

nProgID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the program ID. i.e. The identifier of the program

nType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This method represents the event type. i.e. System error, Startup, Stop, Process failed, Threshold exceeded, Access attempts

rsEventID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event ID, set by EcAgEvent constructor using UUID generator. i.e. The UUID of the event

rsMgmtSvrObjID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the object ID, server object has a unique identifier, UUID (filled in by EcAgManager) when EcAgManager is created by DCE. i.e. The object ID of the EcAgManager of the application.

rsMode

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event mode, i.e. test mode or operational mode. Its string length is at most 8 characters.

rsMsg

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message. i.e. The message of the event

rsTransactionID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction ID. i.e. UUID of the transaction

rsTransactionParentID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction's parent ID. i.e. UUID of the parent transaction

rtTimestamp

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the timestamp of the event.

severity

Privilege: Private

Data Type: EcTAgEventSeverity

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the severity level of the event.

subsys

Privilege: Private

Data Type: EcTAgEventSubsys

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the subsystem for the event.

tupleList

Privilege: Private

Data Type: RWSlistCollectables

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the tuple which makes up the event.

Operations:

```
EcOrderEvent (EcTLong itemID, RWString description,  
requestType type, requestStateType state, EcTime  
timeStateUpdated, requestCost rCost, RWString userId,  
RWString homeDAAC, EcTLong parentId, EcTUR serviceUR)
```

Privilege: Protection Not Identified

No Inheritance

Constructor for the class. The information to be reported to the request tracking server is initialized.

```
ProcessOrderEvent ( )
```

Privilege: Protection Not Identified

No Inheritance

This method takes the information which was set in the class and sends the information to the request tracking server.

`~EcOrderEvent ()`

Privilege: Protection Not Identified

No Inheritance

Default constructor for the class. This method cleans up the service event object by removing any memory allocated for the attributes.

`EcRequest (EcTLong, requestID, RWCString descrip, enum type, EcTTime timeOfLast, EcTLong cpuUtil, EcTLong ioUtil, EcTLong diskUtil, EcTTime rqStart, EcTTime SleepT, EcTTime activeT, EcTTime totalTime)`

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

`~EcRequest ()`

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

`void EcAgEvent ()`

Privilege: Public

Inherited From: EcAgEvent

The default constructor is only used by RWDECLARE. This method is required by Rogue Wave.

`void EcAgEvent (Cat, Typ, Sev, Csci
opt:MgNm, Mg, SSys, App, Prog, Inst, Proc, Trans, TransP)`

Privilege: Public

Inherited From: EcAgEvent

This method represents the constructor. It uses the values in the argument list to set the corresponding attributes.

`void EcAgEvent (const EcAgEvent &event)`

Privilege: Public

Inherited From: EcAgEvent

This method represents the copy constructor.

`EcTVoid Flatten (EcTInt nBufLen, EcTUChar *szBuf)`

Privilege: Public

Inherited From: EcAgEvent

This operation flattens the event.

EcTInt GetAppID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the application ID.

EcTagEventCategory GetCategory ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event category.

EcTInt GetCsci ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the division within subsystem.

RWCString& GetEventID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event ID, set by Event Manager using UUID generator.

EcTInt GetInstanceID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the instanceID is either the application's or the program's Instance ID.

RWCString& GetMgmtSvrObjID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the object ID, server object has a unique identifier, UUID (filled in by Manager) when EcAgManager is created by DCE.

RWCString GetMode ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event mode, i.e. test mode or operational mode.

RWCString& GetMsg ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the message.

EcTInt GetMsgNum ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the message number (related to message).

EcTInt GetNumTuples ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the number of tuples in linked list.

EcTInt GetProcID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the process ID.

EcTInt GetProgID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the program ID.

EcTAgeventSeverity GetSeverity ()

Privilege: Public

Inherited From: EcAgEvent

This operation gets the severity level of the event.

EcTAgeventSubsys GetSubsys ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the subsystem of person creating event.

RWTime& GetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event timestamp.

RWCString & GetTransactionID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction ID.

RWCString& GetTransactionParentID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction's parent ID.

EcAgTuple& GetTuple ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a tuple into the linked list of tuples. It will call Rogue Wave's 'get' function to retrieve an element from the rogue wave linked list.

RWSlistCollectables& GetTupleList ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a list of tuples.

EcTInt GetType ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event type.

EcTVoid InsertTuple (EcAgTuple &tuple)

Privilege: Public

Inherited From: EcAgEvent

This operation inserts a tuple into an event

EcTVoid Restore (EcTInt nBufLen, EcTUChar *szBuf)

Privilege: Public

Inherited From: EcAgEvent

This operation restores from a flattened state.

EcTVoid SetAppID (EcTInt nNewAppID)

Privilege: Public

Inherited From: EcAgEvent

This method sets the application ID.

`EcTInt SetCategory (EcTagEventCategory newCategory)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the category of the event.

`EcTVoid SetCsci (EcTInt nNewCsci)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the division within subsystem.

`EcTVoid SetEventID ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event ID, set by Event Manager using UUID generator.

`EcTVoid SetInstanceID (EcTInt nNewInstanceID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the instance ID.

`EcTVoid SetMgmtSvrObjID (RWCString &rsNewMgmtSvrObjID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the management server object ID.

`EcUtStatus SetMode (RWCString rsNewMode)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the mode of the event.

`EcTVoid SetMsg (RWCString &rsNewMsg)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the event message string.

`EcTVoid SetMsgNum (EcTInt nNewMsgNum)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the message number (related to message).

`EctVoid SetProcID (EctInt nNewProcID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the process ID.

`EctVoid SetProgID (EctInt nNewProgID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the program ID.

`EctInt SetSeverity (EcTagEventSeverity newSeverity)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the severity level of the event.

`EctInt SetSubsys (EcTagEventSubsys newSubsys)`

Privilege: Public

Inherited From: EcAgEvent

this operation sets the subsystem of the event.

`EctVoid SetTimestamp ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event timestamp.

`EctVoid SetTimestamp (RWTime rtNewTime)`

Privilege: Protected Operation

Inherited From: EcAgEvent

This operation sets the event timestamp to a new time.

`EctVoid SetTransactionID (RWCString &rsNewTransactionID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID of the event.

`EctVoid SetTransactionParentID (RWCString
&rsNewTransactionParentID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID for the parent transaction of the event.

`EctVoid SetType (EctInt nNewType)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event type.

`RWSpace binaryStoreSize ()`

Privilege: Public

Inherited From: EcAgEvent

This method returns the number of bytes used by the virtual function `saveGuts (RWFile&)` to store an object.

`friend ostream& operator << (ostream& stream, EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

`friend istream& operator >> (istream& stream, EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

`EcAgEvent& operator= (const EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

This method copies an event.

`EctVoid restoreGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an binary file, using `RWFile`, replacing the previous state.

`EctVoid restoreGuts (RWvistream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an input stream, replacing the previous state.

EcTVoid saveGuts (RWFile& f)

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to a binary file, using class RWFile.

EcTVoid saveGuts (RWvostream& stream)

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to an output stream. virtual

EcTVoid saveGuts (RWvostream&) const {};

void ~EcAgEvent ()

Privilege: Public

Inherited From: EcAgEvent

This method specifies the destructor of the class.

5.11.2.13 Class EcSubOrder

Synopsis:

Parent Class: EcRequest

Is Not A Distributed Object

Is Associated With:

Class: EcSubOrderEvent(Public) generates

Class: EcPfManagedServer(Public) processnon-statechangesub-
orderevents

EcOrder (Aggregation)

EcSubOrder (Aggregation)

Description:

This is a public class which is used by ECS applications to collect resource utilizations associated with sub-order type of request. The class is also used by the application to report the state of the sub-order when the sub-order state changes. Objects in this class should remain until the application has finished processing the associated sub-order request. A sub-order type of request is a child of a hierarchy of sub-orders and services that is associated with a Product Data Order request from an ECS user. At the top of this hierarchy is an order type of request.

Attributes:

archiveUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the total amount of achive utilization which has been collected for this sub-order.

granualFormatList

Privilege: Private

Data Type: GranFormatListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual formats. There is one format list entry for each granual which is associated with this sub-order.

granualList

Privilege: Private

Data Type: GranListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual identifications which are associated with this sub-order.

granualMediaList

Privilege: Private

Data Type: GranMediaListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of media types. There is one media type list entry for each granual which is associated with this sub-order.

granualSizeList

Privilege: Private

Data Type: GranSizeListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual sizes.

mediaCountList

Privilege: Private

Data Type: RWCollectionList(EcTLong)

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains a list of media counts. A sub-order could have more than one type of media being produced from it. This attribute contains the total number of pieces of media which have been produced of each media type.

mediaTypeList

Privilege: Private

Data Type: MediaListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains a list of media types. A sub-order could have more than one type of media being produced from it.

numGranuals

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the number of data granuals which will be produced/retrieved as part of the processing of this sub-order.

parentID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This is the unique request ID of the order or sub-order which spawned the sub-order associated with this object.

shipDateTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is the actual date and time when the products associated with this request were prepared for shipment.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the cpu counter of the system when resource utilization collection is started. This value is then used to calculate the amount of cpu which was used during the collection period.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of disk utilization which has been used while processing this request.

idleTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is a resource utilization counter which contains the approximate amount of real-time that the request has been idle.

ioUtilAtMethodStart

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This attribute is set to the current value of the I/O utilization counter of the system when resource utilization collection is started. This value is then used to calculate the amount of I/O utilization which was used during the collection period.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The running total amount of I/O utilization which has been used while processing this request.

lastEventID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the event identification of the last event that was reported to the MSS event logging capability. This event ID allows an operator to browse through the event log chain for the request in order to show the history of state changes as well as to see any other significant events associated with this request.

requestDate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The date/time at which the request started to be processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

This is the time at which the current state was changed.

type

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequest

The type of the request being processed.

lastEventId

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This is the event identification of the last event that was reported to the MSS event logging capability for this sub-order. This event ID allows the operator to browse through the event log chain for the sub-order in order to show the history of state changes as well as to see any other significant events associated with this sub-order.

Operations:

```
void EcSubOrder (RWCString description, enum type, enum
state, EcTLong parentID, EcTLong numGranuals, GranListType
granualList, GranSizeListType granualSizeList,
GranualFormatListType granFormat, MediaListType
mediaTypeList RWCollectionList mediaCountList)
```

Privilege: Public

No Inheritance

The constructor for the class. The sub-order related information is initialized and the information, including the starting state, are send to the request tracking server. In addition, the resource utilization counters are initialized to zero.

```
EcTVoid GetGranualInfo (EcTLong granualId, enum &granMedia,
RWCString &granFormat, EcTLong &granSize)
```

Privilege: Public

No Inheritance

This method returns the detailed information for a requested granual.

```
EcTVoid GetGranualList (GranListType &granualList)
```

Privilege: Public

No Inheritance

This method returns the list of granual IDs associated with this sub-order.

```
EcTLong GetMediaCount ()
```

Privilege: Public

No Inheritance

This method returns the number of media that have been used when producing the data associated with this sub-order.

```
void ProcessSubOrderEvent (RWCString eventDescription)
```

Privilege: Public

No Inheritance

This method logs an event with the event manager. This allows an ECS application to log a non-state change event associated with the sub-order type of request which is being processed. The method will add signifigant sub-order related information to the event description prior to having the event logged.

`EcTVoid SetGranualSize (EcTLong granualId, EcTLong granualSize)`

Privilege: Public

No Inheritance

This sets the size of a particular granual associated with this sub-order.

`EcTVoid SetMediaCount (EcTInt mediaCount)`

Privilege: Public

No Inheritance

This method sets the number of media that have been used when producing the data associated with this sub-order.

`void ~EcSubOrder ()`

Privilege: Public

No Inheritance

Default destructor for the class. The collected resource utilization and the final state of the sub-order are sent to the request tracking server.

`void EcRequest (RWCString description, enum type, enum state)`

Privilege: Public

Inherited From: EcRequest

This is the constructor for the object. This method sets the attributes of the request object to the passed values and initializes the resource utilization totals to zero.

`EcTLong GetDiskUtilization ()`

Privilege: Public

Inherited From: EcRequest

Returns the current value of the disk utilization attribute.

`EcTLong GetRequestID ()`

Privilege: Public

Inherited From: EcRequest

Returns the value of the request ID attribute.

`trackingStateType GetState ()`

Privilege: Public

Inherited From: EcRequest

Returns the current value of the request state attribute.

`EcTVoid SetDiskUtilization (diskUtilization)`

Privilege: Public

Inherited From: EcRequest

`EcTVoid SetState (state)`

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

`EcTVoid SetState (trackingStateType newState)`

Privilege: Public

Inherited From: EcRequest

This method sets the current value of the request state attribute to the passed value.

`EcTVoid StartCollecting ()`

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and stores them in attributes. This method should be called at the beginning of a method which processes the request associated with this object.

`EcTVoid StopCollecting ()`

Privilege: Public

Inherited From: EcRequest

This method reads from the system the current values of particular resource counters and subtracts from them the associated values stored in the attributes. The resulting value will be added to the running total utilization attribute. This method should be called at the end of a method which processes the request associated with this object.

`void ~EcRequest ()`

Privilege: Public

Inherited From: EcRequest

This is the default destructor of the object. This method cleans up any memory which was allocated to attributes within this object.

ProcessEvent (RWCString eventDescription)

Privilege: Protection Not Identified

Inherited From: EcRequest

No description

EcRequestTracker ()

Privilege: Protection Not Identified

No Inheritance

No description

~EcRequestTracker ()

Privilege: Protection Not Identified

No Inheritance

No description

5.11.2.14 Class EcSubOrderEvent

Synopsis:

Parent Class: EcRequestEvent

Distributed Object

Is Associated With:

Class: EcSubOrder(Public) generates

Class: MsAcTrackingMgr(Private) updatetrackinginfo

Description:

This is a public, distributed object whose purpose is to report information collected about a sub-order type of request. A sub-order type of request is a child of a hierarchy of sub-orders and services that is associated with a Product Data Order request from an ECS user. At the top of this hierarchy is an order type of request. Objects of this class are created with the information to be reported and processed (sent to the request tracking server) and then destroyed. These objects only need to stay around long enough for the event to be processed.

Attributes:

archiveUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the total amount of archive utilization which has been collected for this sub-order.

granualFormatList

Privilege: Private

Data Type: GranFormatListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual formats. There is one format list entry for each granual which is associated with this sub-order.

granualList

Privilege: Private

Data Type: GranListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual identifications which are associated with this sub-order.

granualMediaList

Privilege: Private

Data Type: GranMediaListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of media types. There is one media type list entry for each granual which is associated with this sub-order.

granualSizeList

Privilege: Private

Data Type: GranSizeListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is a list of granual sizes.

mediaCountList

Privilege: Private

Data Type: RWCollectionList(EcTLong)

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains a list of media counts. A sub-order could have more than one type of media being produced from it. This attribute contains the total number of pieces of media which have been produced of each media type.

mediaTypeList

Privilege: Private

Data Type: MediaListType

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains a list of media types. A sub-order could have more than one type of media being produced from it.

numGranuals

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This attribute contains the number of data granuals which will be produced/retrieved as part of the processing of this sub-order.

parentID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

No Inheritance

This is the unique request ID of the order or sub-order which spawned the sub-order associated with this object.

shipDateTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

No Inheritance

This is the actual date and time when the products associated with this request were prepared for shipment.

userId

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

The unique user Identification of the ECS user who submitted the original Product Order request.

activeTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has been actively processed.

cpuUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of cpu processing which has been used while processing this request.

description

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A textual description of the request.

diskUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of disk utilization which has been used while processing this request.

ioUtilization

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The running total amount of I/O utilization which has been used while processing this request.

requestID

Privilege: Private

Data Type: EcTLong

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

A unique identification of the request.

requestStartTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

The date/time at which the request started to be processed.

sleepTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is a resource utilization counter which contains the approximate amount of real-time that the request has not been actively processed.

state

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the current state of the request.

timeOfLastStateUpdate

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the time at which the current state was changed.

totalTime

Privilege: Private

Data Type: EcTTime

Default Value: NOT IDENTIFIED

Inherited From: EcRequestEvent

This is the total amount of real-time which was required to process the request.

type

Privilege: Private
Data Type: enum
Default Value: NOT IDENTIFIED
Inherited From: EcRequestEvent
The type of the request being processed.

category

Privilege: Private
Data Type: EcTAgEventCategory
Default Value: NOT IDENTIFIED
Inherited From: EcAgEvent
This attribute represents the category of the event.

csci

Privilege: Private
Data Type: EcTAgEventCsci
Default Value: NOT IDENTIFIED
Inherited From: EcAgEvent
This attribute represents the division within subsystem. i.e. Identifies the CSCI where the event occurred.

nAppID

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
Inherited From: EcAgEvent
This attribute represents the application ID. i.e. The identifier of the application package

nInstanceID

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
Inherited From: EcAgEvent
This attribute represents the instanceID. It is either the application's or the program's Instance ID.

nMsgNum

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message number (related to message). i.e. Information, Warning, Error, Fatal, Critical

nNumTuples

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the number of tuples in this event.

nProcID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the process ID. i.e. ID of the process which sends out the event

nProgID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the program ID. i.e. The identifier of the program

nType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This method represents the event type. i.e. System error, Startup, Stop, Process failed, Threshold exceeded, Access attempts

rsEventID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event ID, set by EcAgEvent constructor using UUID generator. i.e.The UUID of the event

rsMgmtSvrObjID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the object ID, server object has a unique identifier, UUID (filled in by EcAgManager) when EcAgManager is created by DCE. i.e.The object ID of the EcAgManager of the application.

rsMode

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the event mode, i.e. test mode or operational mode. Its string length is at most 8 characters.

rsMsg

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the message. i.e.The message of the event

rsTransactionID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction ID. i.e.UUID of the transaction

rsTransactionParentID

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the transaction's parent ID. i.e.UUID of the parent transaction

rtTimestamp

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the timestamp of the event.

severity

Privilege: Private

Data Type: EcTagEventSeverity

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the severity level of the event.

subsys

Privilege: Private

Data Type: EcTagEventSubsys

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the subsystem for the event.

tupleList

Privilege: Private

Data Type: RWSlistCollectables

Default Value: NOT IDENTIFIED

Inherited From: EcAgEvent

This attribute represents the tuple which makes up the event.

Operations:

```
EcSubOrderEvent (EcTLong itemID, RWString description,  
requestType type, requestStateType state, EcTTime  
timeStateUpdated, requestCost rCost, long parentID,  
granListType granualList, granListSizeType granualSizeList,  
mType mediaType, long mediaCount, date shipDateTime)
```

Privilege: Protection Not Identified

No Inheritance

Constructor for the class. The information to be reported to the request tracking server is initialized.

```
EcTVoid ProcessSubOrderEvent ()
```

Privilege: Public

No Inheritance

This method takes the information which was set in the class and sends the information to the request tracking server.

```
void ~EcSubOrderEvent ()
```

Privilege: Public

No Inheritance

Default constructor for the class. This method cleans up the sub-order event object by removing any memory allocated for the attributes.

```
EcRequest (EcTLong, requestID, RWCString descrip, enum type,  
EcTTime timeOfLast, EcTLong cpuUtil, EcTLong ioUtil, EcTLong  
diskUtil, EcTTime rqStart, EcTTime SleepT, EcTTime activeT,  
EcTTime totalTime)
```

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

```
~EcRequest ()
```

Privilege: Protection Not Identified

Inherited From: EcRequestEvent

No description

```
void EcAgEvent ()
```

Privilege: Public

Inherited From: EcAgEvent

The default constructor is only used by RWDECLARE. This method is required by Rogue Wave.

```
void EcAgEvent (Cat,Typ,Sev,Csci  
opt:MgNm,Mg,SSys,App,Prog,Inst,Proc,Trans, TransP)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the constructor. It uses the values in the argument list to set the corresponding attributes.

```
void EcAgEvent (const EcAgEvent &event)
```

Privilege: Public

Inherited From: EcAgEvent

This method represents the copy constructor.

```
EcTVoid Flatten (EcTInt nBufLen, EcTUChar *szBuf)
```

Privilege: Public

Inherited From: EcAgEvent

This operation flattens the event.

```
EcTInt GetAppID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the application ID.

```
EcTAgEventCategory GetCategory ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event category.

```
EcTInt GetCsci ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the division within subsystem.

```
RWCString& GetEventID ()
```

Privilege: Public

Inherited From: EcAgEvent

This method gets the event ID, set by Event Manager using UUID generator.

`EcTInt GetInstanceID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the instanceID is either the application's or the program's Instance ID.

`RWCString& GetMgmtSvrObjID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the object ID, server object has a unique identifier, UUID (filled in by Manager) when EcAgManager is created by DCE.

`RWCString GetMode ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the event mode, i.e. test mode or operational mode.

`RWCString& GetMsg ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the message.

`EcTInt GetMsgNum ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the message number (related to message).

`EcTInt GetNumTuples ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the number of tuples in linked list.

`EcTInt GetProcID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the process ID.

`EcTInt GetProgID ()`

Privilege: Public

Inherited From: EcAgEvent

This method gets the program ID.

EcTagEventSeverity GetSeverity ()

Privilege: Public

Inherited From: EcAgEvent

This operation gets the severity level of the event.

EcTagEventSubsys GetSubsys ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the subsystem of person creating event.

RWTime& GetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event timestamp.

RWCString & GetTransactionID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction ID.

RWCString& GetTransactionParentID ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the transaction's parent ID.

EcAgTuple& GetTuple ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a tuple into the linked list of tuples. It will call Rogue Wave's 'get' function to retrieve an element from the rogue wave linked list.

RWSlistCollectables& GetTupleList ()

Privilege: Public

Inherited From: EcAgEvent

This method gets a list of tuples.

EcTInt GetType ()

Privilege: Public

Inherited From: EcAgEvent

This method gets the event type.

`EcTVoid InsertTuple (EcAgTuple &tuple)`

Privilege: Public

Inherited From: EcAgEvent

This operation inserts a tuple into an event

`EcTVoid Restore (EcTInt nBufLen, EcTUChar *szBuf)`

Privilege: Public

Inherited From: EcAgEvent

This operation restores from a flattened state.

`EcTVoid SetAppID (EcTInt nNewAppID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the application ID.

`EcTInt SetCategory (EcTagEventCategory newCategory)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the category of the event.

`EcTVoid SetCsci (EcTInt nNewCsci)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the division within subsystem.

`EcTVoid SetEventID ()`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event ID, set by Event Manager using UUID generator.

`EcTVoid SetInstanceID (EcTInt nNewInstanceID)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the instance ID.

`EcTVoid SetMgmtSvrObjID (RWCString &rsNewMgmtSvrObjID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the management server object ID.

EcUtStatus SetMode (RWCString rsNewMode)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the mode of the event.

EcTVoid SetMsg (RWCString &rsNewMsg)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the event message string.

EcTVoid SetMsgNum (EcTInt nNewMsgNum)

Privilege: Public

Inherited From: EcAgEvent

This method sets the message number (related to message).

EcTVoid SetProcID (EcTInt nNewProcID)

Privilege: Public

Inherited From: EcAgEvent

This method sets the process ID.

EcTVoid SetProgID (EcTInt nNewProgID)

Privilege: Public

Inherited From: EcAgEvent

This method sets the program ID.

EcTInt SetSeverity (EcTAgEventSeverity newSeverity)

Privilege: Public

Inherited From: EcAgEvent

This operation sets the severity level of the event.

EcTInt SetSubsys (EcTAgEventSubsys newSubsys)

Privilege: Public

Inherited From: EcAgEvent

this operation sets the subsystem of the event.

EcTVoid SetTimestamp ()

Privilege: Public

Inherited From: EcAgEvent

This method sets the event timestamp.

`EcTVoid SetTimestamp (RWTime rtNewTime)`

Privilege: Protected Operation

Inherited From: EcAgEvent

This operation sets the event timestamp to a new time.

`EcTVoid SetTransactionID (RWCString &rsNewTransactionID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID of the event.

`EcTVoid SetTransactionParentID (RWCString
&rsNewTransactionParentID)`

Privilege: Public

Inherited From: EcAgEvent

This operation sets the transaction ID for the parent transaction of the event.

`EcTVoid SetType (EcTInt nNewType)`

Privilege: Public

Inherited From: EcAgEvent

This method sets the event type.

`RWSpace binaryStoreSize ()`

Privilege: Public

Inherited From: EcAgEvent

This method returns the number of bytes used by the virtual function `saveGuts (RWFile&)` to store an object.

`friend ostream& operator << (ostream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`friend istream& operator >> (istream& stream, EcAgEvent&
event)`

Privilege: Public

Inherited From: EcAgEvent

`EcAgEvent& operator= (const EcAgEvent& event)`

Privilege: Public

Inherited From: EcAgEvent

This method copies an event.

`EcTVoid restoreGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an binary file, using RWFile, replacing the previous state.

`EcTVoid restoreGuts (RWvistream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method reads an object's state from an input stream, replacing the previous state.

`EcTVoid saveGuts (RWFile& f)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to a binary file, using class RWFile.

`EcTVoid saveGuts (RWvostream& stream)`

Privilege: Public

Inherited From: EcAgEvent

This method writes an object's state to an output stream. virtual

`EcTVoid saveGuts (RWvostream&) const {};`

`void ~EcAgEvent ()`

Privilege: Public

Inherited From: EcAgEvent

This method specifies the destructor of the class.

5.11.2.15 Class ILMItemB

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: ILMMgrB(Private) accessesforinfo

Class: ILMItemB(Public) ismadeupof

Description:

The ILM Item is the asset in the ECS inventory or held as an ECS supply (consumable or non-consumable) which is managed and/or maintained.

Attributes:

ECSbarcodeData

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives the ECS bar-coded data.

barcodeData

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives all the data that is stored on the ECS/local bar coding.

basisOfIssue

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the unit(s) by/in which the item is counted (each, dozen, by the poihund, by the box, etc.).

componentOf

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is a component or subassembly of a higher order assembly.

condition

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

UIndicates the current ocondition of the item.

cost

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the original purchase cost of the item.

custodian

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates who is the assigned custodian for the asset/item.

dateReceived

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the date that the item was received.

description

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives the description of the item.

disposalBasis

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates if the item has any special disposal or turn-in instructions.

documentation

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates if the item is documentation and gives its designation.

equipmentControlled

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is under any special controls (such as for high value or hazardous items).

equipmentItemTracked

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the equipment item has any special system tracking such as maintenance up-time/down-time tracking (special interest).

hwSwType

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is hardware or software.

invTypeItem

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the type of inventory to which the item belongs (equipment item, line replaceable unit, consumable item, spare part).

itemIDnumberBarcoded

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is bar-coded with an identification number.

itemIdentification

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Unique alphanumeric code used to designate the item.

license

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether or not a software item is under license.

licenseAgreement

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is under license from a vendor (especially for software).

localBarcodeData

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates any locally assigned bar-code data.

location

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the location where the item is stored (designation locally specified).

maintainer

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates who is assigned to perform maintenance on the asset/item.

maintenanceAgreement

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether item is under a maintenance agreement with the vendor or a third-party vendor.

manufacturer

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the name of the item manufacturer.

modelNo

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives the model number of the item.

owner

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates who owns the item (Government/contractor; ECD project/ other).

partNo

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives the part number of the item.

serialNo

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Gives the serial number of the item.

serviceLife

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item has a known or anticipated service life, such as battery shelf life.

softwareType

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

For software items, indicates what type of software (operating system, application, utility, etc.).

source

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates where the item came from originally (such as the vendor's name).

status

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates the current status of the item.

trainingMaterial

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item is used for training purposes.

version

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

For software, whether the item is issued in versions or has a version number.

versionNo

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

For software, the version number assigned by the issuer.

warrantyExpiration

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates when the warranty on the item expires.

warrantyList

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the item covered on a warranty list.

Operations:

void AddItem ()

Privilege: Public

No Inheritance

This operation adds an item to the inventory.

void RemoveItem ()

Privilege: Public

No Inheritance

This operation removes an item from the inventory.

void UpdateItem ()

Privilege: Public

No Inheritance

This operation updates the information about an item in the inventory.

5.11.2.16 Class MsAcAddress

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
MsAcUsrProfile (Aggregation)

Description:

This class contains address, phone and Fax information for an ECS user.

Attributes:

city

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

country

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

fax

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

phone

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

state

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

street1

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

street2

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

zip

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

Operations:

const RWCString GetCity ()

Privilege: Public
No Inheritance

const RWCString GetCountry ()

Privilege: Public
No Inheritance

const RWCString GetFax ()

Privilege: Public
No Inheritance

const RWCString GetPhone ()

Privilege: Public
No Inheritance

const RWCString GetStreet1 ()

Privilege: Public
No Inheritance

```
const RWCString GetStreet2 ()
    Privilege: Public
    No Inheritance

const RWCString GetZip ()
    Privilege: Public
    No Inheritance

void MsAcAddress ()
    Privilege: Public
    No Inheritance

EcTInt SetCity (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetCountry (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetFax (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetPhone (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetStreet1 (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetStreet2 (const RWCString)
    Privilege: Public
    No Inheritance

EcTInt SetZip (const RWCString)
    Privilege: Public
    No Inheritance
```

```
void ~MsAcAddress ()
```

Privilege: Public

No Inheritance

5.11.2.17 Class MsAcUserName

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

MsAcUsrProfile (Aggregation)

Description:

This class contains user name information for an ECS user

Attributes:

firstName

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

lastName

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

middleInit

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

title

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Operations:

```
void GetFirstName ()
```

Privilege: Public

No Inheritance

```
void GetLastName ()
```

Privilege: Public

No Inheritance

```

void GetMiddleInit ()
    Privilege: Public
    No Inheritance

void GetTitle ()
    Privilege: Public
    No Inheritance

void MsAcUserName ()
    Privilege: Public
    No Inheritance

void SetFirstName (RWCString)
    Privilege: Public
    No Inheritance

void SetLastName (RWCString)
    Privilege: Public
    No Inheritance

void SetMiddleInit (RWCString)
    Privilege: Public
    No Inheritance

void SetTitle (RWCString)
    Privilege: Public
    No Inheritance

void ~MsAcUserName ()
    Privilege: Public
    No Inheritance

```

5.11.2.18 Class MsAcUsrProfile

Synopsis:

```

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: MsTtHTMLItems(Private) Createsandreads
Class: MsAcUsrProfileMgr(Public) Populates
Class: MsTtHTMLMenu(Private) createsandreads
Class: MsAcUsrProfileMgr(Public) manages
Class: MsAcTrackingMgr(Private) updateaccountbalance
MsAcRegUser (Aggregation)

```

Description:

This class contains profile information for an ECS user

Attributes:

PI

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

accountBalance

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

accountNumber

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

affiliation

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

altMailAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

altShipAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

billAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

creationDate

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

emailAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

expirationDate

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

homeDAAC

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

mailAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

mediaPref

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

organization

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

privilegeLevel

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

projectName

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

researchFiled

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

shipAddr

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

sponsor

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

telNum

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

userId

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

userName

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

Operations:

EcTLong accountBalance GetAccountBalance ()

Privilege: Public

No Inheritance

void GetAccountNumber ()

Privilege: Public

No Inheritance

void GetAffiliation ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrCity ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrFax ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrPhone ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrState ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrStreet1 ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrStreet2 ()
 Privilege: Public
 No Inheritance

void GetAltMailAddrZip ()
 Privilege: Public
 No Inheritance

void GetAltMailCountry ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrCity ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrFax ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrPhone ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrState ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrStree2 ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrStreet1 ()
 Privilege: Public
 No Inheritance

void GetAltShipAddrZip ()
 Privilege: Public
 No Inheritance

void GetAltShipCountry ()
 Privilege: Public
 No Inheritance

void GetAltShipState ()
 Privilege: Public
 No Inheritance

void GetBillAddrCity ()
 Privilege: Public
 No Inheritance

void GetBillAddrCountry ()
 Privilege: Public
 No Inheritance

void GetBillAddrFax ()
 Privilege: Public
 No Inheritance

void GetBillAddrPhone ()
 Privilege: Public
 No Inheritance

void GetBillAddrState ()
 Privilege: Public
 No Inheritance

void GetBillAddrStreet1 ()
 Privilege: Public
 No Inheritance

void GetBillAddrStreet2 ()
 Privilege: Public
 No Inheritance

void GetBillAddrZip ()
 Privilege: Public
 No Inheritance

void GetCreationDate ()
 Privilege: Public
 No Inheritance

void GetEmailAddress ()
 Privilege: Public
 No Inheritance

void GetExpirationDate ()
 Privilege: Public
 No Inheritance

void GetHomeDAAC ()
 Privilege: Public
 No Inheritance

void GetMailAddrCity ()
Privilege: Public
No Inheritance

void GetMailAddrCountry ()
Privilege: Public
No Inheritance

void GetMailAddrFax ()
Privilege: Public
No Inheritance

void GetMailAddrPhone ()
Privilege: Public
No Inheritance

void GetMailAddrState ()
Privilege: Public
No Inheritance

void GetMailAddrStreet1 ()
Privilege: Public
No Inheritance

void GetMailAddrStreet2 ()
Privilege: Public
No Inheritance

void GetMailAddrZip ()
Privilege: Public
No Inheritance

void GetMailAddress ()
Privilege: Public
No Inheritance

void GetMediaPref ()
Privilege: Public
No Inheritance

void GetOrganization ()
 Privilege: Public
 No Inheritance

void GetPIFirstName ()
 Privilege: Public
 No Inheritance

void GetPILastName ()
 Privilege: Public
 No Inheritance

void GetPIMiddleInit ()
 Privilege: Public
 No Inheritance

void GetPITitle ()
 Privilege: Public
 No Inheritance

void GetPrivilegeLevel ()
 Privilege: Public
 No Inheritance

void GetProjectName ()
 Privilege: Public
 No Inheritance

void GetResearchField ()
 Privilege: Public
 No Inheritance

void GetShipAddrCity ()
 Privilege: Public
 No Inheritance

void GetShipAddrCountry ()
 Privilege: Public
 No Inheritance

void GetShipAddrFax ()
 Privilege: Public
 No Inheritance

void GetShipAddrPhone ()
 Privilege: Public
 No Inheritance

void GetShipAddrState ()
 Privilege: Public
 No Inheritance

void GetShipAddrStreet1 ()
 Privilege: Public
 No Inheritance

void GetShipAddrStreet2 ()
 Privilege: Public
 No Inheritance

void GetShipAddrZip ()
 Privilege: Public
 No Inheritance

void GetSponsor ()
 Privilege: Public
 No Inheritance

void GetTelNum ()
 Privilege: Public
 No Inheritance

void GetUserFirstName ()
 Privilege: Public
 No Inheritance

void GetUserId ()
 Privilege: Public
 No Inheritance

```
void GetUserLastName ()
    Privilege: Public
    No Inheritance

void GetUserMiddleInit ()
    Privilege: Public
    No Inheritance

void GetUserTitle ()
    Privilege: Public
    No Inheritance

void MsUserProfile (EcTVoid)
    Privilege: Public
    No Inheritance

EcTVoid SetAccountBalance (EcTLong newBalance)
    Privilege: Public
    No Inheritance

void SetAccountNumber (RWCString)
    Privilege: Public
    No Inheritance

void SetAffiliation (RWCString)
    Privilege: Public
    No Inheritance

void SetAltMailAddrCity (RWCString)
    Privilege: Public
    No Inheritance

void SetAltMailAddrFax (RWCString)
    Privilege: Public
    No Inheritance

void SetAltMailAddrPhone (RWCString)
    Privilege: Public
    No Inheritance
```

void SetAltMailAddrState (RWCString)

Privilege: Public

No Inheritance

void SetAltMailAddrStreet1 (RWCString)

Privilege: Public

No Inheritance

void SetAltMailAddrStreet2 (RWCString)

Privilege: Public

No Inheritance

void SetAltMailAddrZip (RWCString)

Privilege: Public

No Inheritance

void SetAltMailCountry (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrCity (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrFax (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrPhone (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrState (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrStree2 (RWCString)

Privilege: Public

No Inheritance

void SetAltShipAddrStreet1 (RWCString)
Privilege: Public
No Inheritance

void SetAltShipAddrZip (RWCString)
Privilege: Public
No Inheritance

void SetAltShipCountry (RWCString)
Privilege: Public
No Inheritance

void SetAltShipState (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrCountry (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrFax (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrPhone (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrState (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrStreet1 (RWCString)
Privilege: Public
No Inheritance

void SetBillAddrZip (RWCString)
Privilege: Public
No Inheritance

void SetBillAddtCity (RWCString)

Privilege: Public

No Inheritance

void SetBillAddtStreet2 (RWCString)

Privilege: Public

No Inheritance

void SetCreationDate (const RWDate)

Privilege: Public

No Inheritance

void SetEmailaddress (RWCString)

Privilege: Public

No Inheritance

void SetExpirationDate (const RWDate)

Privilege: Public

No Inheritance

void SetHomeDAAC (RWCString)

Privilege: Public

No Inheritance

void SetMailAddrCity (RWCString)

Privilege: Public

No Inheritance

void SetMailAddrCountry (RWCString)

Privilege: Public

No Inheritance

void SetMailAddrFax (RWCString)

Privilege: Public

No Inheritance

void SetMailAddrPhone (RWCString)

Privilege: Public

No Inheritance

void SetMailAddrState (RWCString)
Privilege: Public
No Inheritance

void SetMailAddrStreet1 (RWCString)
Privilege: Public
No Inheritance

void SetMailAddrStreet2 (RWCString)
Privilege: Public
No Inheritance

void SetMailAddrZip (RWCString)
Privilege: Public
No Inheritance

void SetMailAddress (RWCString)
Privilege: Public
No Inheritance

void SetMediaPref (RWCString)
Privilege: Public
No Inheritance

void SetOrganization (RWCString)
Privilege: Public
No Inheritance

void SetPIFirstName (RWCString)
Privilege: Public
No Inheritance

void SetPILastName (RWCString)
Privilege: Public
No Inheritance

void SetPIMiddleInit (RWCString)
Privilege: Public
No Inheritance

void SetPITitle (RWCString)
 Privilege: Public
 No Inheritance

void SetPrivilegeLevel (RWCString)
 Privilege: Public
 No Inheritance

void SetProjectName (RWCString)
 Privilege: Public
 No Inheritance

void SetResearchField (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrCity (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrCountry (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrFax (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrPhone (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrState (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrStreet1 (RWCString)
 Privilege: Public
 No Inheritance

void SetShipAddrStreet2 (RWCString)

Privilege: Public

No Inheritance

void SetShipAddrZip (RWCString)

Privilege: Public

No Inheritance

void SetSponsor (RWCString)

Privilege: Public

No Inheritance

void SetTelNum (RWCString)

Privilege: Public

No Inheritance

void SetUserFirstName (RWCString)

Privilege: Public

No Inheritance

void SetUserId (RWCString)

Privilege: Public

No Inheritance

void SetUserLastName (RWCString)

Privilege: Public

No Inheritance

void SetUserMiddleInit (RWCString)

Privilege: Public

No Inheritance

void SetUserTitle (RWCString)

Privilege: Public

No Inheritance

void ~MsUserProfile (EcTVoid)

Privilege: Public

No Inheritance

5.11.2.19 Class MsAcUsrProfileMgr

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
Class: MsAcTrackingMgr(Private)
Class: MsAcUsrProfile(Public) Populates
Class: MsTtHTMLItems(Private) Requestuserprofilefrom
Class: MsRgUIMgrB(Private) handlesrequestforM&Ouserprofile
Class: MsAcManager(Private) manages
Class: MsAcUsrProfile(Public) manages
Class: MsAcUserProfile(Private) populates
Class: MsTtHTMLMenu(Private) requestuserprofilefrom
Class: MsBaBAASManagerB(Private) updateavailablebalance
Class: MsAcTrackingMgr(Private) updatesuserprofile
Class: MsAcManagerUI(Private) uses
Class: MsAcRegUserMgr(Private) uses

Description:

This class represents the User Profile Manager class that governs the update and maintenance of information in the MsAcUsrProfile class. An ECS science user's available balance will be retrieved using this class and be debited by the amount of each data product request received by MSS.

Attributes:

Operations:

EctVoid DeleteProfile ()

Privilege: Public
No Inheritance

EctVoid InsertProfile ()

Privilege: Public
No Inheritance

void MsAcUserProfileMgr ()

Privilege: Public
No Inheritance

EctVoid ReplicateProfileToSMC ()

Privilege: Public
No Inheritance

EcTVoid RetrieveProfile (RWCString userId)

Privilege: Public
No Inheritance

EcTVoid RetrieveProfile (RWCString lastname, firstName,
middleInitial)

Privilege: Public
No Inheritance

EcTVoid RetrieveProfile (RWCString accountNumber)

Privilege: Public
No Inheritance

EcTVoid RetrieveProfileList ()

Privilege: Public
No Inheritance

EcTVoid UpdateProfile ()

Privilege: Public
No Inheritance

void ~MSAcUserProfileMgr ()

Privilege: Public
No Inheritance

5.11.2.20 Class MsAcUsrRequest

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: MsAcUsrRequestMgr(Private) manages
Class: MsAcManagerUI(Private) uses

Description:

Attributes:

PI

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

accountNumber

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

affiliation

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

billAddr

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

emailAddr

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

expirationDate

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

homeDAAC

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

mailAddr

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

mediaPref

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

operator

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

organization

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

processDate

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

projectName

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

requestDate

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

researchFiled

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

shipAddr

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

sponsor

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

status

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

telNum

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

userName

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

usrRequestId

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance

userReqId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

Operations:

GetAccountNumber ()

Privilege: Protection Not Identified
No Inheritance

GetAffiliation ()

Privilege: Protection Not Identified
No Inheritance

GetBillAddrCity ()

Privilege: Protection Not Identified
No Inheritance

GetBillAddrCountry ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrFax ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrPhone ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrState ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrStreet1 ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrStreet2 ()
Privilege: Protection Not Identified
No Inheritance

GetBillAddrZip ()
Privilege: Protection Not Identified
No Inheritance

GetEmailAddress ()
Privilege: Protection Not Identified
No Inheritance

GetExpirationDate ()
Privilege: Protection Not Identified
No Inheritance

GetHomeDAAC ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrCity ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrCountry ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrFax ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrPhone ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrState ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrStreet1 ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrStreet2 ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddrZip ()
Privilege: Protection Not Identified
No Inheritance

GetMailAddress ()
Privilege: Protection Not Identified
No Inheritance

GetMediaPref ()
Privilege: Protection Not Identified
No Inheritance

GetOperator ()
Privilege: Protection Not Identified
No Inheritance

GetOrganization ()
Privilege: Protection Not Identified
No Inheritance

GetPIFirstName ()
Privilege: Protection Not Identified
No Inheritance

GetPILastName ()
Privilege: Protection Not Identified
No Inheritance

GetPIMiddleInit ()
Privilege: Protection Not Identified
No Inheritance

GetPITitle ()
Privilege: Protection Not Identified
No Inheritance

GetProcessDate ()
Privilege: Protection Not Identified
No Inheritance

GetProjectName ()
Privilege: Protection Not Identified
No Inheritance

GetRequestDate ()
Privilege: Protection Not Identified
No Inheritance

GetResearchField ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrCity ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrCountry ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrFax ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrPhone ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrState ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrStreet1 ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrStreet2 ()
Privilege: Protection Not Identified
No Inheritance

GetShipAddrZip ()
Privilege: Protection Not Identified
No Inheritance

GetSponsor ()
Privilege: Protection Not Identified
No Inheritance

GetStatus ()
Privilege: Protection Not Identified
No Inheritance

GetTelNum ()
Privilege: Protection Not Identified
No Inheritance

GetUserFirstName ()
Privilege: Protection Not Identified
No Inheritance

GetUserLastName ()
Privilege: Protection Not Identified
No Inheritance

GetUserMiddleInit ()
Privilege: Protection Not Identified
No Inheritance

GetUserTitle ()
Privilege: Protection Not Identified
No Inheritance

GetUsrRequestId ()
Privilege: Protection Not Identified
No Inheritance

MsAcUsrRequest ()
Privilege: Protection Not Identified
No Inheritance

SetAccountNumber (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetAffiliation (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrCountry (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrFax (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrPhone (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrState (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrStreet1 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddrZip (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddtCity (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetBillAddtStreet2 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetEmailaddress (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetExpirationDate (const RWDate)
Privilege: Protection Not Identified
No Inheritance

SetHomeDAAC (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrCity (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrCountry (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrFax (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrPhone (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrState (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrStreet1 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrStreet2 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddrZip (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMailAddress (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetMediaPref (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetOperator (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetOrganization (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetPIFirstName (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetPILastName (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetPIMiddleInit (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetPITitle (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetProcessDate (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetProjectName (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetRequestDate (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetResearchField (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrCity (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrCountry (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrFax (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrPhone (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrState (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrStreet1 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrStreet2 (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetShipAddrZip (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetSponsor (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetStatus (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetTelNum (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetUserFirstName (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetUserLastName (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetUserMiddleInit (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetUserTitle (RWCString)
Privilege: Protection Not Identified
No Inheritance

SetUstRequestId (RWCString)
Privilege: Protection Not Identified
No Inheritance

~MsAcUsrRequest ()
Privilege: Protection Not Identified
No Inheritance

EctVoid GetStatus (RWCString userReqId)
Privilege: Public
No Inheritance

EctVoid GetUserRequest (MsAcUserRequest)
Privilege: Public
No Inheritance

EctVoid GetUserRequestId (RWCString userReqId)
Privilege: Public
No Inheritance

`EcTVoid MsAcUserRequest ()`

Privilege: Public
No Inheritance

`EcTVoid SetStatus (RWCString userReqId)`

Privilege: Public
No Inheritance

`EcTVoid SetUerRequestId (RWCString userReqId)`

Privilege: Public
No Inheritance

`EcTVoid SetUserRequest (MsAcUserRequest)`

Privilege: Public
No Inheritance

`EcTVoid ~MsAcUserRequest ()`

Privilege: Public
No Inheritance

5.11.2.21 Class MsAgMonitor

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

Class: MsAgPerfEvent(Private) createdby

Class: MsAgPerfEvent(Private) creates

Class: MsAgPortMonitor(Private) receivedrequestsfrom

Class: MsAgPortMonitor(Private) receivesrequestsfrom

Class: EcAgManager(Private) sendsdatato

Class: MsAgPortMonitor(Private) sendsrequeststo

Class: MsAgEventManager(Private) talksto

Description:

MsAgMonitor is spawned by the MsAgSubagent. It provides the local polling capability to monitor resources being managed. This can avoid the costly remote polling done by management applications while still being able to monitor the state of the resources. If error conditions occurred, it informs the agent to send a trap to the management application(s). The scope of this local polling is the host. The time interval applies to all the resources that are monitored by this object. This class also provides the capability for monitoring transient processes. Applications may specify that transient processes that they

create be monitored for their presence until the monitoring is no longer required. Start and stop indications are needed from applications. Between the start and stop requests, the transient process is monitored. In the event that a monitored process fails, a notification is sent to the interested application.

Attributes:

pBindingVector

Privilege: Private

Data Type: MsAgMgmtBindingVector*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a binding vector.

pTblMgr

Privilege: Private

Data Type: MsAgTblMgr*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a table manager.

Operations:

EcUtStatus CheckProcessState (EcTInt pid)

Privilege: Private

No Inheritance

This operation checks the process state.

EcUtStatus CheckStatus (EcTInt index)

Privilege: Private

No Inheritance

this operation checks the status.

EcUtStatus CheckThresholds (MsAgTePerf *pEntry,
MsAgMgmtHandle *pMgrHdle)

Privilege: Private

No Inheritance

This operation checks the thresholds.

void MsAgMonitor (MsAgTblMgr *pTblMgr, MsAgMgmtBindingHandle
*pMgmtBindingHandle)

Privilege: Public

No Inheritance

This operation is the constructor for the MsAgMonitor class.

EcUtStatus PollPerformance ()

Privilege: Public

No Inheritance

This operation polls for performance metrics.

EcTVoid PollPerformanceTable (MsAgSNMPTbl *pTable)

Privilege: Private

No Inheritance

This operation polls performance table.

EcUtStatus PollStatus ()

Privilege: Public

No Inheritance

This method checks if all processes registered are still running.

void ~MsAgMonitor ()

Privilege: Public

No Inheritance

This method represents the constructor of the object.

5.11.2.22 Class MsAgSubAgent

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

Class: MsAgSubAgent(Public)

Class: MsAgScheduler(Private) accesses

Class: ManagementFramework(Private) communicateswith

Class: MsAgAgent(Private) communicateswith

Class: MsAgScheduler(Private) schedulesfor

Class: ManagementFramework(Private)

setthresholdsandpollingintervals

Class: MsAgAppMIB(Private) usedby

Class: MsAgScheduler(Private) usedby

Class: EcAgProxy(Public) uses

Class: MsAgAppMIB(Private) uses

Description:

This managed object class supports SNMP MIB extensions. It receives requests from the master agent. Based on Get or Set requests, it performs the retrieval or set functions onto resource or resource managers using available API. This object will instantiate another object MsAgMonitor to perform local polling on resources on the host.

Attributes:

pBuffer

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a StaticBuffer.

pDepGate

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a deputy gate.

pDiscoverer

Privilege: Private

Data Type: MsAgDiscoverer*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a discoverer.

pEventManager

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to an event manager.

pMonitor

Privilege: Private

Data Type: MsAgMonitor*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a monitor.

pPortMonitor

Privilege: Private

Data Type: MsAgPortMonitor*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a port monitor.

pSugAgentCfg

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to the subagent configuration.

pTblMgr

Privilege: Private

Data Type: MsAgTblMgr*

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a table manager.

pThread

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute represents a pointer to a thread.

nDiscoverIndex

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

discover index into scheduler

nDiscoverInterval

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

discover interval

nPerformanceIndex

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

performance index into scheduler

nPerformanceInterval

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
performance interval

nStatusIndex

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
status index into scheduler

nStatusInterval

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
status polling interval

pDeputyGate

Privilege: Private
Data Type: MsAgDeputyGate*
Default Value: NOT IDENTIFIED
No Inheritance

pLock

Privilege: Private
Data Type: DCEPthreadMutex*
Default Value: NOT IDENTIFIED
No Inheritance
DCE thread mutex

pRegistry

Privilege: Private
Data Type: MsAgRegistry*
Default Value: NOT IDENTIFIED
No Inheritance
registry pointer

pScheduler

Privilege: Private

Data Type: MsAgScheduler*

Default Value: NOT IDENTIFIED

No Inheritance

scheduler pointer

pSubAgentConfig

Privilege: Private

Data Type: MsAgSubAgentConfig*

Default Value: NOT IDENTIFIED

No Inheritance

Operations:

void MsAgSubAgent ()

Privilege: Public

No Inheritance

This method represents the constructor of the object.

static pthread_addr_t SchdThreadExec (pthread_addr_t pThread)

Privilege: Private

No Inheritance

This method spawns a DCE thread.

void ~MsAgSubAgent ()

Privilege: Public

No Inheritance

This method represents the destructor of the object.

EcUtStatus ErrorHandler (MSTagSubAgentTaskType taskType,
EcTInt nNewInterval)

Privilege: Private

No Inheritance

performs error handling for PerformTask

EcUtStatus PerformTask (MSTagSubAgentTaskType taskType,
EcTInt nNewInterval=-1)

Privilege: Public

No Inheritance

Removes entry from schedule which would poll using the old interval, and insert an entry into the schedule with the new interval. OR Stop repeating specified task. OR tells discoverer to discover all applications or tell monitor to poll performance/status immediately and one time only OR tell monitor to poll performance after specified time interval and only once

5.12 Production Planning Classes

5.12.1 Production Planning Classes Overview

The Production Planning Subsystem shares a number of objects with the Data Processing Subsystem through the PDPS database. These objects allow these two subsystems to keep track of processing jobs and statistics. In addition, Release B adds a public object called PIONdemandPRNB, which allows the Data Server Subsystem to submit on-demand production requests.

5.12.2 Production Planning Class Descriptions

5.12.2.1 Class PIAlternateNB

Synopsis:

Parent Class: PIDataTypeReq
Is Not A Distributed Object
Is Associated With:
This class is derived from the class PIDataTypeReq

Description:

This class defines an alternate input to a PGE. It is used for PGE that has multiple inputs, and allows a secondary choice if on (or more) or its primary inputs are not available.

Attributes:

myDefaultOrder

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
Default priority of an alternate input data type.

myDefaultTimer

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
The default amount of time that the Subscription Manager will wait for an alternate input data type to arrive.

myPrimary

Privilege: Private
Data Type: PIDataTypeReq
Default Value: NOT IDENTIFIED
No Inheritance
This points to the primary input for which this input is an alternate.

myTemporalFlag

Privilege: Private

Data Type: EcTBoolean

Default Value: NOT IDENTIFIED

No Inheritance

Indicates (if TRUE) whether the most recent alternate input data granule can be used instead of the primary data granule of the same data type, if a primary data granule can't be found for the production request time frame.

myWaitFor

Privilege: Private

Data Type: EcTBoolean

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates whether or not Planning should wait for this alternate input after the alternate inputs timer has expired. If the value is set to false, then Planning will allow the PGE to be executed without this input.

myCommandString

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This string is used to construct the acquire command for the data type, used to stage the data from the data server.

myDataTypeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This is the identified of the Data Type that is required.

myDataTypeReq

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This attribute specifies the data type requirement of a PGE. The string specifies the select statement to be applied to the Data Granule table, to determine the input Data Granules for a given Data Processing Request.

myInspectString

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This attribute stores the command string sent to the Data Server to inspect fields of the metadata.

myLogicalID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

The logical id relates to the PGE input identifier for a particular product type. This attribute is required within the SDP toolkit process control interface.

myNumNeeded

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This represents the number of inputs required by the PGE associated with PIDataTypeReq.

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This is the identifier of the PGE that requires this data.

myQAThreshold

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIDataTypeReq

This attribute describes the quality threshold to be applied to the data granules to "approve" their suitability for production before releasing a scheduled PGE. The threshold is specified in terms of the ESDT parameter list of the data type.

myType

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: PlDataTypeReq

This represents the type of input: Required, primary, or backup.

Operations:

void PlAlternate ()

Privilege: Public

No Inheritance

This is the constructor for the class.

void ~PlAlternate ()

Privilege: Public

No Inheritance

This is the destructor for the class.

EctVoid PlDataTypeReq ()

Privilege: Public

Inherited From: PlDataTypeReq

Default constructor for the class

EctVoid PlDataTypeReq (RWCString:PgeId,
RWCString:DataTypeId)

Privilege: Public

Inherited From: PlDataTypeReq

This constructor takes in the PGE ID and Data Type ID of the Data Type Required class to be created.

EctVoid ~PlDataTypeReq ()

Privilege: Public

Inherited From: PlDataTypeReq

This is the destructor for the class.

5.12.2.2 Class PICluster

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PITile(Public)

Class: PITileScheduledNB(Public)

Description:

This class defines the Cluster of tiles used by the PGE. A cluster is set up to allow a group of tiles to be scheduled at one time that use the same or similar geographic inputs.

Attributes:

`myLastOrbit`

Privilege: Private

Data Type: `EctInt`

Default Value: NOT IDENTIFIED

No Inheritance

The number of the orbit that contains the last data granule in this cluster.

`myNumTiles`

Privilege: Private

Data Type: `EctInt`

Default Value: NOT IDENTIFIED

No Inheritance

The number of tiles for this cluster.

`myPredictedAvailability`

Privilege: Private

Data Type: `RWTime`

Default Value: NOT IDENTIFIED

No Inheritance

The predicted availability time of all data granules in this cluster.

Operations:

`void PlCluster ()`

Privilege: Public

No Inheritance

This is the constructor for the class.

`void PlCluster (EctInt Tiles, RWTime PredAvail, EctInt LastOrbit)`

Privilege: Public

No Inheritance

This constructor takes in a LastOrbit, a PredictedAvail, and a NumTiles and fills in the attributes when the class is created.

`EctInt ReturnLastOrbit ()`

Privilege: Public

No Inheritance

This returns the value in myLastOrbit.

EctInt ReturnNumTiles ()

Privilege: Public

No Inheritance

This method returns myNumTiles.

RWTime ReturnPredictedAvail ()

Privilege: Public

No Inheritance

This method returns the value in myPredictedAvailability.

EctInt Update (RWCString Attribute, RWCString Value)

Privilege: Public

No Inheritance

This method updates the specified Attribute with the specified Value. If the attribute does not exist, or the value is out of range a "bad" return code is returned.

void ~PlCluster ()

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.3 Class PIDPRB

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: DpPrScheduler(Private)

Class: PIPGE(Public)

Class: PIProdStratNB(Private) basesprioritieson

Class: PIONdemandPRNB(Public) composes

Class: PIPGE(Public) generates

Class: PIDataGranule(Public) input

Class: PIPGEActivity(Private) isaplanned

Class: PIProductionRequestB(Public) isbrokenupintoindividual

Class: PIProductionRequestB(Public) ismadeupof

Class: PIResourceRequirement(Public) isrequiredby

Class: PITimer(Private) isusedtosettimeronalternateinputs

Class: PIDataGranule(Public) matchescorresponding

Class: PIDataGranule(Public) output

Class: DpPrScheduler(Private) schedules

Class: DpPrScheduler(Private) schedulesinprocessing

Class: PIDataGranule(Public) uses

PIDPRs (Aggregation)

Description:

This class describes an individual run of a PGE.

Attributes:

`myActualStartTime`

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

The actual time that this DPR began processing in DPS

`myBaselineTime`

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

The time the DPR is predicted to begin in that latest baseline plan.

`myCompletionState`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Status indicator describing active status of data processing request

`myDprId`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Unique identifier for the DPR instance

`myInputDataInstanceList`

Privilege: Private

Data Type: RWTValslist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

List describing all the input files required within for the PGE

myNextConditionalDPR

Privilege: Private

Data Type: PIDPR&

Default Value: NOT IDENTIFIED

No Inheritance

For PGEs with mode-based activations, a reference to the next DPR associated with this one. For normal PGEs, this will be null for the first DPR in the PR list (PIProductionRequest::myDPRs).

myOutputDataInstanceList

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

List describing all the output files to be produced by the PGE

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The id of the PGE that is being executed in this DPR

myPredictedStartTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

myPriority

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Priority for the data processing request is inherited from the production request, but may be modified individually

myProductionRequestId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The id of the Production Request of which this DPR is a part.

myStartTime

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

myStopTime

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

myUR

Privilege: Private

Default Value: NOT IDENTIFIED

No Inheritance

The Universal Reference generated by Data Server for the product generated by this processing request.

Operations:

EctVoid Cancel ()

Privilege: Public

No Inheritance

Cancel the DPR within the data processing job scheduler

Boolean CheckAvailability ()

Privilege: Public

No Inheritance

Checks to see if all the data dependencies for a DPR have been fulfilled.
Returns true or false.

EctVoid CheckPredictedAvailabilityNB ()

Privilege: Public

No Inheritance

Determines the predicted availability of the input data granules

RWCString GetCommandString (RWCString:DataType)

Privilege: Public

No Inheritance

Operation to format the command string that is sent to Data Server for a Query.

RWTValslist<RWCString> GetInputGranuleList ()

Privilege: Public

No Inheritance

Read the list of required data granules from the PDPS Database for the PGE to be used.

void GetLogicalId (RWCString:DataType)

Privilege: Public

No Inheritance

Operation to used to get the logical Id of the Data Type used for this production request

RWCString GetMyPGEType ()

Privilege: Public

No Inheritance

This operation will get the PGE Type in order to determine whether or not a PGE is a tiling PGE.

RWTime GetMyTimer ()

Privilege: Public

No Inheritance

Returns the length of time a timer should wait for the primary input data after receiving the required inputs before using the backups.

RWTValslist<RWCString> GetOutputGranuleList ()

Privilege: Public

No Inheritance

Read the list of date granules that the Pge will generate from the PDPS Database.

EctVoid Modify ()

Privilege: Public

No Inheritance

Modify the DPR within the data processing job scheduler. The attributes of the DPR that may have been modified are priority and the anticipated start / stop times of the DPR used to set the alarms in the data processing scheduler.

PlDPR PlDPR (RWCString:DprId)

Privilege: Public

No Inheritance

Constructor

EctVoid Release ()

Privilege: Public

No Inheritance

Releases the DPR within the data processing job scheduler. The DPR is released when the data dependencies of that DPR have been fulfilled.

EctVoid RescheduleNB ()

Privilege: Public

No Inheritance

This operation is called to replace a failed DPR with an OnDemand DPR to allow continuation of a chain of processing. This operation will call the ReplaceDprJob operation in DpPrScheduler.

EctVoid Schedule ()

Privilege: Public

No Inheritance

schedule the DPR within the data processing job scheduler

EctVoid Status ()

Privilege: Public

No Inheritance

status the DPR within the data processing job scheduler

void ~PlDPR ()

Privilege: Public

No Inheritance

Destructor method

void PlDPR ()

Privilege: Public

No Inheritance

Constructor method

5.12.2.4 Class PIDPRs

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

None

Description:

This is a collector class for the PIDPR class, and contains methods to select Data Processing Requests from the PDPS database and to iterate through them. This class may be implemented by a suitable Rogue Wave template class.

Attributes:**Operations:**

```
void Add (PIDPR: dpr)
    Privilege: Public
    No Inheritance
    Add a DPR to the collection
```

```
void Delete (PIDPR: dpr)
    Privilege: Public
    No Inheritance
    Delete an DPR from the collection.
```

```
PIDPR First ()
    Privilege: Public
    No Inheritance
    Returns the first DPR within the collection
```

```
PIDPR Next ()
    Privilege: Public
    No Inheritance
    Returns the next DPR within the collection (or NULL if no more
    activities).
```

```
PIDPRs PIDPRs ()
    Privilege: Public
    No Inheritance
    Constructor method
```

```
void SelectDPRs (Interval)
    Privilege: Public
    No Inheritance
    Builds the collection for all DPRs within the plan that lies within the
    time interval specified
```

```
void ~PIDPRs ()
    Privilege: Public
    No Inheritance
    Destructor method
```

5.12.2.5 Class PIDataGranule

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PISubMsgCb(Private) Notificationofarrivalofgranule
Class: PISourcetoDsHistoryNB(Private) UpdatesArrivalTime
Class: PIDPRB(Public) input
Class: PIDPRB(Public) matchescorresponding
Class: PIDPRB(Public) output
Class: PIDataTypeB(Public) populates
Class: PIFile(Private) simulatesstroageof
Class: PIDATRecord(Private) updatesavailabilityof
Class: PIDATRecord(Private) updatestheavailabilityof
Class: PIDPRB(Public) uses

Description:

This class describes individual instances or granules of data types.

Attributes:

myActualAvailability

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
Date and time that data was made available to subscription manager.

myAvailability

Privilege: Private
Data Type: EcTBoolean
Default Value: NOT IDENTIFIED
No Inheritance
Flag to indicate availability of data

myBaselineTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
The availability time of the data granule according to the baselined active plan agreed to amongst DAACs requiring the data granule.

myDataGranuleId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
ID of data granule.

myDataTypeId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
The identifier of the data type of the data granule.

myESDTParmVals

Privilege: Private
Data Type: GIParameterList
Default Value: NOT IDENTIFIED
No Inheritance
Selected metadata fields associated to the data type required to determine suitability in the production (such as quality info or geophysical attributes)

myPredictedAvailability

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
Predicted time at which the data will be available in ECS, used to determine PGE schedule estimates.

myPredictedStagingTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
This is an ROUGH estimate of the amount of time that this file will take to stage, based on the size of the file, the age of the file (determines whether its in deep storage or on a local disk) and whether the file was remote or local.

myStartTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
The start time and date of the data

myStopTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
The stop time and date of the data

myTileId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
For data granules that are geographical tiles instead of time continuous data, this attribute will note which tile within the start time and stop time of the orbit set (an orbit set covers the entire earth once) this data granule matches. Tile attributes, including the latitude and longitude of the coordinates that bound this tile, are found in the PGE Profile for PGEs that produce this data type. For data granules that are not tiles, this attribute would be set to zero.

myUR

Privilege: Private
Data Type: EcUrUR
Default Value: NOT IDENTIFIED
No Inheritance
Universal Reference by which to reference the granule within the data server.

Operations:

void Create (RWTime:start, RWTime:stop)

Privilege: Public
No Inheritance
Create a entry in the DBMS if this is a unique instance of the granule.

`void DeleteDummyGranule (RWCString:myDataGranuleId)`

Privilege: Public

No Inheritance

This operation is used for PGEs that produce spatial tiles to delete that dummy data granule that represents the last data granule which will be collected (or processed for higher level products, assuming the data granules are produced in pretty much the same order that they are collected). The Subscription Manager uses the notification that this data has arrived to begin to query the Data Server for all the actual data granules that are needed as input to produce this tile.

`PlDPRs FindAssociatedDPRs ()`

Privilege: Public

No Inheritance

Method to determine the data processing requests associated to the data granule

`EcTBoolean GetAvailability ()`

Privilege: Public

No Inheritance

Returns whether the data item is available within the ECS or not.

`PlDataGranule PlDataGranule ()`

Privilege: Public

No Inheritance

Constructor method

`RWTime PredictStagingTime ()`

Privilege: Public

No Inheritance

Computes the value stored in myPredictedStagingTime.

`void RegisterAvailability (EcUrUR:instUR, GlParameterList:
instESDTParmVals)`

Privilege: Public

No Inheritance

Method to register that a data instance that was predicted has arrived, and to record the UR and metadata associated to that data.

`void ~PlDataGranule ()`

Privilege: Public
No Inheritance
Destructor method

`DeleteGranule (RWCString:myDataGranuleId)`

Privilege: Protection Not Identified
No Inheritance
This operation will delete the predicted granule created by the production request editor for tiling.

5.12.2.6 Class PlDataTypeB

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: DsClQuery(Private)
Class: DsClSubscription(Private)
Class: PlDataSource(Private) checks
Class: PIPGE(Public) input
Class: IoAdProductSearchCommand(Private)
isusedtointerfacetothetheAdvertisingServiceby
Class: PIDATRecord(Private) matches
Class: PIPGE(Public) output
Class: PlDataGranule(Public) populates
Class: PlSubscriptionSubmitIF(Private) updates
PlDataTypeCatalogue (Aggregation)
PIPGEProfile (Aggregation)

Description:

This class describes a data type known to the planning subsystem. This is a description of an input or output type, distinct to a granule or instance of the data type. The class is an abstraction or proxy that describes one of the Data Server ESDTs. The class captures data and operations that are required to subscribe and receive notification from the Data Server when a new instance of the Data Type arrives.

Attributes:

`myArchiveCenter`

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
DAAC where Data Type is archived.

myCatalogueCategory

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

Catalogue category indicates whether the Data Type is valid for a production request, which would indicate it is generated from a PGE at a site, otherwise the Data Type is some intermediate file, or input file received from another site.

myDServURString

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Universal Reference to identify Data Server providing services (retrieve, insert, inspect) for the Data Type.

myDataTypeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the Data Type.

myDescription

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute provides a text description for the Data Type

myDynamicFlag

Privilege: Private

Data Type: EcTBoolean

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether the Data Type is dynamic or static. Examples of dynamic are L0, L1 data sets etc. with a frequent update time. Examples of static are calibration files which only change with a new version of a PGE.

myESDTParmList

Privilege: Private

Data Type: GIParameterList

Default Value: NOT IDENTIFIED

No Inheritance

A parameter list used within the inspect to the Data Server, to retrieve the metadata associated to a newly arrived instance of the Data Type.

myInstrumentName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Instrument name associated with PGE

myName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Attribute describes the ESDT Name for the data set within the Data Server.

myNominalSize

Privilege: Private

Data Type: EcTFloat

Default Value: NOT IDENTIFIED

No Inheritance

Nominal size of the data type

myProcessingCenter

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the name of the DAAC that produces this data type.

myProvider

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

the DAAC that is maintaining the data

myQASubscription

Privilege: Private

Data Type: EcTBoolean

Default Value: NOT IDENTIFIED

No Inheritance

Captures whether a subscription has been set up for the QA of this data type.

mySatelliteName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Satellite name associated to PGE

myService

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The service (usually ACQUIRE) that serves the data

mySpatialFlag

Privilege: Private

Data Type: EcTBoolean

Default Value: EcDFalse

No Inheritance

If this flag is set, then the data granules for this type require spatial coordinates to define them as well as start and stop time of the data. This flag will identify data that has been placed into tiles.

mySubscriptionFlag

Privilege: Private

Data Type: EcTBoolean

Default Value: EcDFalse

No Inheritance

Set to true if Planning currently has a subscription on this data type with the Data Server or Ingest

myUsedByCenter

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

List of DAACs which use this Data Type as input for their processing.

Used for InterDAAC planning.

Operations:

EcTVoid FindDataAvailability (RWTime Start,RWTime Stop)

Privilege: Public

No Inheritance

Ensures that the Data Granules that are required to fulfill a production request are captured in the PDPS database

IoAdService * GetAdService (EcUtStatus &)

Privilege: Public

No Inheritance

Used to get the advertisement title for a service for this data type. This advertisement title will be used when submitting a subscription.

EcTVoid InspectDataArrival ()

Privilege: Public

No Inheritance

Creates a UR from the notification message received and uses the UR to extract the required metadata (ESDT parameter list) from the Data Server.

EcTVoid InstallReceiver ()

Privilege: Public

No Inheritance

Install a receiving queue in which to receive notification of subscription from the Data Server.

EcTVoid MatchDataArrival ()

Privilege: Public

No Inheritance

Match the data arrival with the predictions within Data Granule table.

PlDataType PlDataType ()

Privilege: Public

No Inheritance

This is the default constructor for the class.

`PlDataType PlDataType (RWCString:DataTypeId)`

Privilege: Public

No Inheritance

This constructor takes in the ID of the Data Type to be created.

`EcUtStatus QueryDATRecords (RWTime: StartTime, RWTime: StopTime, RWTlistSval<PlDATRecords>)`

Privilege: Public

No Inheritance

This operation returns a list of DAT records for this data type for the specified interval.

`EcTVoid RegisterDataArrival ()`

Privilege: Public

No Inheritance

Procedure to be called on notification of data arrival, this a control procedure that manages the inspection of the data, comparison to the data instances, and then subsequent handling of associated DPRs to the arrived data.

`EcUtStatus Submit ()`

Privilege: Private

No Inheritance

Used to submit a subscription for this data type

`EcUtStatus Withdraw ()`

Privilege: Public

No Inheritance

Used to withdraw a subscription for this data type

`void ~PlDataType ()`

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.7 Class PlDataTypeReq

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PlMetaDataChecks(Public)

Class: PlMetaDataChecks(Public) performs

Description:

This class contains the data that associates a PGE to an input data type.

Attributes:

`myCommandString`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This string is used to construct the acquire command for the data type, used to stage the data from the data server.

`myDataTypeId`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identified of the Data Type that is required.

`myDataTypeReq`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute specifies the data type requirement of a PGE. The string specifies the select statement to be applied to the Data Granule table, to determine the input Data Granules for a given Data Processing Request.

`myInspectString`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute stores the command string sent to the Data Server to inspect fields of the metadata.

`myLogicalID`

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

The logical id relates to the PGE input identifier for a particular product type. This attribute is required within the SDP toolkit process control interface.

myNumNeeded

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This represents the number of inputs required by the PGE associated with PlDataTypeReq.

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the PGE that requires this data.

myQAThreshold

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute describes the quality threshold to be applied to the data granules to "approve" their suitability for production before releasing a scheduled PGE. The threshold is specified in terms of the ESDT parameter list of the data type.

myType

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

This represents the type of input: Required, primary, or backup.

Operations:

EctVoid PlDataTypeReq ()

Privilege: Public

No Inheritance

Default constructor for the class

EctVoid PlDataTypeReq (RWCString:PgeId,
RWCString:DataTypeId)

Privilege: Public

No Inheritance

This constructor takes in the PGE ID and Data Type ID of the Data Type Required class to be created.

EctVoid ~PlDataTypeReq ()

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.8 Class PLErrorAction

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PlProductionReuquest(Private)

PIPGEProfile (Aggregation)

Description:

This class provides for the definition of actions based on the value of the PGE return code. It can create an AutoSys alarm with specified text, or it can place the associated DPR into planning for execution.

Attributes:

myAction

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

This is the action to be taken when this PGE returns the value in myStatus. It is an enumeration type with the values of (none, sendalarm, submitDPR).

myAlarmText

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This defines the text that will appear in the AutoSys Alarm window for the value in myStatus.

myDPRID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the DPR that is to be executed when the value in myStatus is returned by the PGE.

myStatus

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This is the return code value (the status) for which the action is to be taken.

myTimeframeDataType

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute defines the Data Type from which the TimeFrame for the new DPR (if the action is SubmitDPR).

Operations:

void Get (EcTInt PGEID, EcTInt ExitCode)

Privilege: Public

No Inheritance

This method retrieves the attributes of this class from the PDPS database.

EcTInt HandleError (EcTInt ExitCode, EcTInt DPRID)

Privilege: Public

No Inheritance

This method takes in the ExitCode and the DPRID of the failed DPR and takes the actions specified in myAction. If the action is SendAlarm, the DpPrCOTSMgr is called with the text defined in myAlarmText. If the action is SubmitDPR, then PLOnDemandMgr is called with the PIProductionRequest linked to this class.

void PLErrorAction ()

Privilege: Public

No Inheritance

This is the default constructor for the class.

void PLErrorAction (EcTInt Status, enum Action, RWCString AlarmText, EcTInt DPRID)

Privilege: Public

No Inheritance

This constructor takes as arguments values for the class attributes and populates them.

`void Put (EctInt PGEID)`

Privilege: Public

No Inheritance

This method puts the attributes stored in the class into the PDPS database.

`void ~PlErrorAction ()`

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.9 Class **PIGroundEvent**

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PIResourceRequirement(Public) hasa

Class: PIRpResourceReservation(Private)

inthePDPSdatabaseisupdatedby

Class: PIGroundActivity(Private) isaplanned

Class: DpPrScheduler(Private) isscheduledinprocessingby

Class: PIGroundActivity(Private) whenplannedis

PIGroundEvents (Aggregation)

Description:

This class describes a Ground Event which is recorded in the PDPS database. A Ground Event marks the allocation of resources to some none-production task such as maintainance.

Attributes:

`myDescription`

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

This attribute provides storage for a text description of a ground event

`myDuration`

Privilege: Private

Data Type: Time

Default Value: NOT IDENTIFIED

No Inheritance

This attribute describes the duration of the Ground Event

myName

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

Event name entered by Resource Manager for this ground event.

myPriority

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

This attribute describes the priority of the Ground Event.

myTemplateFlag

Privilege: Private

Data Type: Boolean

Default Value: False

No Inheritance

This attribute describes whether the ground event is to be saved as a template description of a ground events. Ground Events with this attribute True will be retained in the PDPS database. Those with this attribute as False will automatically be deleted a month after the completion date of the Ground Event.

myWinEndTime

Privilege: Private

Data Type: Time

Default Value: NOT IDENTIFIED

No Inheritance

This attribute describes the end time of the window of opportunity for when the Ground Event may be planned

myWinStartTime

Privilege: Private

Data Type: Time

Default Value: NOT IDENTIFIED

No Inheritance

This attribute describes the start time of the window of opportunity for when the Ground Event may be planned

Operations:

`void Cancel ()`

Privilege: Public

No Inheritance

Cancel the Ground Event within the data processing job scheduler

`void Create ()`

Privilege: Public

No Inheritance

Create an ground event in the PDPS database

`void Delete ()`

Privilege: Public

No Inheritance

Delete the ground event from the PDPS database

`void Modify ()`

Privilege: Public

No Inheritance

Modify the Ground Event within the data processing job scheduler. The attributes of the Ground Event that may have been modified are priority and the anticipated start / stop times of the DPR used to set the alarms in the data processing scheduler.

`PlGroundEvent PlGroundEvent ()`

Privilege: Public

No Inheritance

Constructor method

`void Schedule ()`

Privilege: Public

No Inheritance

Schedule the Ground Event within the data processing job scheduler

`void Status ()`

Privilege: Public

No Inheritance

Status the Ground Event within the data processing job scheduler

`void ~PlGroundEvent ()`

Privilege: Public

No Inheritance

Destructor method

5.12.2.10 Class PIMetaDataChecks

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PlDataTypeReq(Public)

Class: PlDataTypeReq(Public) performs

Description:

This class provides a metadata field and its corresponding value to be checked against the actual metadata of the specified input data granule when deciding if a particular input should be used, or a PGE should be executed.

Attributes:

`myDataTypeId`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the data type for which the defined metadata values should be checked.

`myMetaDataOper`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the operation (=, >, <) for the comparison of the metadata field.

`myMetaDataParmName`

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the name of the metadata parameter to be compared.

myMetaDataType

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the type (int, string, etc...) of the metadata parameter to be compared.

myMetaDataValue

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the value for the metadata parameter to be compared with the actual metadata value.

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the PGE for which the metadata values are to be checked.

Operations:

void CheckForCondition (RWCString, PlDataGranule &, EcTBoolean &)

Privilege: Public

No Inheritance

This method checks the metadata for the specified Data Granule for the metadata field and value.

void CompareActualWithCondition (EcTFloat, EcTFloat, RWCString &, EcUtStatus &)

Privilege: Public

No Inheritance

This method compares the Float metadata attribute value with the value of the actual file's metadata.

void CompareActualWithCondition (RWCString &, RWCString &, RWCString &, EcUtStatus &)

Privilege: Public

No Inheritance

This method compares the String metadata attribute value with the actual file's metadata.

```
void CompareActualWithCondition (EcTInt, EcTInt, RWCString &,
EcUtStatus &)
```

Privilege: Public

No Inheritance

This method compares the Int metadata attribute value with the actual file's metadata.

5.12.2.11 Class PIONdemandPRNB

Synopsis:

Parent Class: PIProductionRequestB

Distributed Object

Is Associated With:

Class: EcRequest(Private) ProvidesStatus

Class: PIONdemandManagerNB(Private) aremanagedby

Class: PIReplanCriteria(Private) checkforneededforreplannotification

Class: PIRescUseThreshNB(Private) comparewith

Class: PIDPRB(Public) composes

Class: PIPGE(Public) requests

Description:

This class is the specialization class which holds additional/modified attributes and operations needed for On-Demand production requests.

Attributes:

myFailedDprID

Privilege: Private

Data Type: RWCString

Default Value: Null

No Inheritance

myDPRsNB

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

This list is a list of DPRs generated by a production request.

myDataCollectionStartTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

Start time for the production request.

myDataCollectionStopTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

Stop time for the production request.

myDescription

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

A textual description of the production request.

myNumDPRsToKeep

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

For intermittently activated PGEs, this indicates the number of DPRs to be kept for the PR. For normal PGEs, the value is 0.

myNumDPRsToSkip

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

For intermittently activated PGEs, this indicates the number of DPRs to skip (after a number, specified in myNumDPRstoKeep, are created). For normal PGEs, this value is 0.

myOutputDataType

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

Identifies the product desired by the ECS User.

myPGEIdentifier

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIProductionRequestB

Identifies the PGE

myPRCollectionNB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
Pointer to my parent PIPRCollectorNB

myPRTypeNB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
Indicates production type: routine, on-demand, or reprocessing.

myPriority

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
User requested priority for the submitted production request.

myRequestId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
A unique identifier for this production request.

myRequesterId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
Identifies the person entering the production request.

myTargetDate

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
The desired time that a production request will be completed - for informational purposes only.

myUserTypeNB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB

myUsrParaList

Privilege: Private
Data Type: RWTValSlist<RWCString>
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
A list of PGE parameters for this production request.

myUsrParaValueList

Privilege: Private
Data Type: RWTValSlist<RWCString>
Default Value: NOT IDENTIFIED
Inherited From: PIProductionRequestB
A list of parameter values for the PGE associated with this production request.

Operations:

PlOnDemandPRNB PlOnDemandPRNB ()

Privilege: Public
No Inheritance
Constructor method

EcUrUR ReturnUR ()

Privilege: Public
No Inheritance
This operation is called by the Data Server to initiate action to retrieve the UR for the product being generated by this request. It calls the SubmitSubscription routine in PlOnDemandManagerNB, and then after the subscription is returned and the cleanup is accomplished it returns the UR of the product.

EcTInt ValidateOPR ()

Privilege: Public
No Inheritance
Operation to determine if thresholds are exceeded and if data is available to generate product. If conditions are satisfactory then request is submitted to processing. Status is returned in form of an enumerated type; 1 = Submitted, 2 = Delayed, and 3 = Exceeded.

`void ~PlOnDemandPRNB ()`

Privilege: Public
No Inheritance
Destructor method.

`EctFloat CalculateResourceUsageNB (enum)`

Privilege: Public
Inherited From: PlProductionRequestB
Determines the resource usage of a Production Request by multiplying the number of DPRs needed to satisfy this request with the PlResourceRequirement and PlPerformance values in PlPGEProfile for the PGE in the request.

`void DefinePGERuns ()`

Privilege: Public
Inherited From: PlProductionRequestB
Defines the DPRs corresponding to the PGE.

`void Modify (RWCString:Modfield, RWCString:ModVal)`

Privilege: Public
Inherited From: PlProductionRequestB
Modifies the production request.

`void PlProductionRequest ((RWCString:Product, RWCString:PGE,
RWTime:Start, RWTime:Stop, ECTInt:Prior))`

Privilege: Public
Inherited From: PlProductionRequestB
Constructor for the PlProductionRequest class.

`void RetrieveAllProdReq ()`

Privilege: Public
Inherited From: PlProductionRequestB
Retrieves all the production requests currently in effect.

`void StoreProductionRequest (PlProductionRequestB)`

Privilege: Public
Inherited From: PlProductionRequestB
Stores the production request in the PDPS database.

RequestCompleted (UR)

Privilege: Protection Not Identified

No Inheritance

Operation to accept UR of a completed product and associate with a given Production Request, then remove it from the list of requests kept by the Manager and return the UR to Data Server via the Callback routine.

5.12.2.12 Class PLOutputYield

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

None

Description:

Specifies the recipe to describe the output data granules for a PGE.

Attributes:

myCommandString

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This string is used to construct the insert command for the data type, used to destage the data to the data server.

myDataTypeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier for the Data Type for which this is the output yield.

myLogicalID

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

The logical id relates to the PGE output identifier for a particular product type. This attribute is required within the SDP toolkit process control interface.

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier of the PGE that this Output Yield is for.

myYield

Privilege: Private

Data Type: EcTFloat

Default Value: NOT IDENTIFIED

No Inheritance

Describes the number of data granules produced from the Data Processing Request. These are assumed to be evenly distributed in time across the acquisition processing time of the DPR.

Operations:

void PlOutputYield ()

Privilege: Public

No Inheritance

This is the default constructor for the class.

void PlOutputYield (RWCString:PgeId, RWCString:DataTypeId)

Privilege: Public

No Inheritance

This constructor takes in the PGE ID and Data Type ID for the created output yield object.

void ~PlOutputYield ()

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.13 Class PIPGE

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PIDPRB(Public)

Class: PIPgeFactory(Private) createdby

Class: PIPgeFactory(Private) creates

Class: PIDPRB(Public) generates

Class: PIDataTypeB(Public) input

Class: PIResourceRequirement(Public) isreferencedby
Class: PIUserParameters(Public) isreferencedby
Class: PIDataTypeB(Public) output
Class: PIONdemandPRNB(Public) requests
Class: PIProductionRequestB(Public) requests
PIPGEProfile (Aggregation)

Description:

This is the base class within a generalization heirachy that describes PGEs. The class defines abstract operations required for the planning subsystem to work out when a PGE needs to be scheduled. As well as containing the key attributes defining the PGE.

Attributes:

myDefaultTimer

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This is the default timer value for the PGE for use in Alternative Input selection.

myInputDataTypeList

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

List of input data types needed by the PGE

myInstrument

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Instrument for which the PGE is appropriate

myNumCPUs

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

This is the number of CPUs that the PGE requires. It should be 1 for non-parallel processing PGEs.

myOutputDataTypeList

Privilege: Private

Data Type: RWTVaISlist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

List of output data types needed by the PGE

myPGENAME

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Name of the PGE

myPGEVersion

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Version number of the PGE

myPlatform

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Platform for which the PGE is appropriate, may be a list

myTestOperational

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

No Inheritance

Indicates if the PGE's status is test or operational

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the unique identifier of the PGE.

myType

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
This is the type of the PGE.

myNormalStagingTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
This is the time to stage the input data (during normal processing) for this PGE.

myReprocessingStagingTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
This is the time to stage the input data of the PGE when performing Reprocessing.

Operations:

void Delete ()

Privilege: Public
No Inheritance
Delete the PGE from the PDPS database

void FindDataAvailability (Interval)

Privilege: Public
No Inheritance
This is used to find what input data is available for this PGE.

void GenerateDPRs (PlProductionRequest)

Privilege: Public
No Inheritance
Generate the Data Processing Requests to fulfill the Production Request.

void Modify (TestOrOper:enum)

Privilege: Public
No Inheritance
This is used to modify the attributes of this class.

PlPGE PlPGE ()

Privilege: Public
No Inheritance
Constructor method

PlPGE (int PGEid)

Privilege: Protection Not Identified
No Inheritance
Construct the PGE from the PDPS database

void UpdateVersion ()

Privilege: Public
No Inheritance
Update the version of the PGE

void ~PlPGE ()

Privilege: Public
No Inheritance
Destructor method

PlPGE PlPGE (RWCString:PGEid)

Privilege: Public
No Inheritance
This constructor takes the PGE ID of the object to be constructed.

5.12.2.14 Class PIPGECollection

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PlInstrumentModes(Private)

Description:

This class handles collections of PGEs. One or more PGEs may be run with different instrument modes. This collection allows multiple PGE profiles for those modes.

Attributes:

Operations:

5.12.2.15 Class PIPGEProfile

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PIInstrumentModes(Private)

Description:

This class describes the collection of information that describes a PGE to the Planning subsystem.

Attributes:

Operations:

```
void $PlPGEProfile (...)
```

Privilege: Public
No Inheritance
Constructor function

```
void DeletePGEProfile (PGEProfID:int)
```

Privilege: Public
No Inheritance
Deletes a given PGE profile

```
void ModifyPGEProfile (PGEProfID:int, ModType:enum,  
ModField:String, ModValue:String)
```

Privilege: Public
No Inheritance
Modifies a given PGE profile

```
void RetrievePGEProfile (PlPGEProfID)
```

Privilege: Public
No Inheritance
Retrieves a given PGE profile

```
void ~PlPGEProfile (...)
```

Privilege: Public
No Inheritance
Destructor function

5.12.2.16 Class PIPerformance

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PIResourceRequirement(Public)
PIPGEProfile (Aggregation)

Description:

This class describes the performance statistics of a PGE. These performance statistics are established at AI&T. The class also contains attributes to describe the statistics updated from the Data Processing subsystem.

Attributes:

myElapsedTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
Elapsed time for the PGE during AI&T

myMaxMemoryUse

Privilege: Private
Data Type: EcTFloat
Default Value: NOT IDENTIFIED
No Inheritance
Maximum memory required by the PGE during AI&T

myNoOfBlockInOper

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
Number of input blocks for the PGE during AI&T

myNoOfBlockOutOper

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
Number of output blocks for the PGE during AI&T

myNoOfPageFaults

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Number of PGE page faults for the PGE during AI&T

myNoOfSwaps

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Number of PGE swaps for the PGE during AI&T

myPGECPUTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

CPU required for the PGE during AI&T

myPgeId

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This is the unique identifier of the PGE.

myRunElapsedTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

Elapsed time for the PGE during production

myRunMaxMemoryUse

Privilege: Private

Data Type: EcTFloat

Default Value: NOT IDENTIFIED

No Inheritance

Maximum memory required by the PGE during production

myRunNoOfBlockInOper

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Number of input blocks for the PGE during production

myRunNoOfBlockOutOper

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

The number of output blocks used by the PGE during production.

myRunNoOfPageFaults

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Number of PGE page faults for the PGE during production

myRunNoOfSwaps

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

Number of PGE swaps for the PGE during production

myRunPGUCPUTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

CPU required for the PGE during production

myRunSharedMemoryUse

Privilege: Private

Data Type: EcTFloat

Default Value: NOT IDENTIFIED

No Inheritance

Shared memory required by the PGE during production

mySharedMemoryUse

Privilege: Private

Data Type: EcTFloat

Default Value: NOT IDENTIFIED

No Inheritance

Shared memory required by the PGE during AI&T

Operations:

void PlPerformance ()

Privilege: Public

No Inheritance

This is the default constructor for the class.

void PlPerformance (RWString:PgeId)

Privilege: Public

No Inheritance

This is the constructor for the class that takes Pgeid as an argument.

EcTVoid UpdateRunTimePerfPar (EcTInt:ParId, RWString:ParVal)

Privilege: Public

No Inheritance

This routine updates the runtime performance parameters. It is called after a PGE has been run to record the latest performance information.

void ~PlPerformance ()

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.17 Class PIProductionRequestB

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PIProductionRequestUI(Private) displays

Class: PIDPRB(Public) isbrokenupintoindividual

Class: PIDPRB(Public) ismadeupof

Class: PIPGE(Public) requests

Class: PIPlanGenerationUIB(Private) selectsforplanning

PIPRCollectionNB (Aggregation)

Description:

This class is the instructions describing the order for data set(s) to be produced. A production request typically specifies a request for a Data Set to be produced for an extended period of time (e.g a month's worth of some product). There are three types of production requests: Standard, Re-processing and On-Demand.

Attributes:

myDPRsNB

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

No Inheritance

This list is a list of DPRs generated by a production request.

myDataCollectionStartTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

Start time for the production request.

myDataCollectionStopTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

Stop time for the production request.

myDescription

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

A textual description of the production request.

myNumDPRsToKeep

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

For intermittently activated PGEs, this indicates the number of DPRs to be kept for the PR. For normal PGEs, the value is 0.

myNumDPRsToSkip

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

For intermittently activated PGEs, this indicates the number of DPRs to skip (after a number, specified in myNumDPRstoKeep, are created). For normal PGEs, this value is 0.

myOutputDataType

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Identifies the product desired by the ECS User.

myPGEIdentifier

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Identifies the PGE

myPRCollectionNB

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Pointer to my parent PIPRCollectorNB

myPRTypeNB

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

Indicates production type: routine, on-demand, or reprocessing.

myPriority

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

No Inheritance

User requested priority for the submitted production request.

myRequestId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
A unique identifier for this production request.

myRequesterId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
Identifies the person entering the production request.

myTargetDate

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
The desired time that a production request will be completed - for informational purposes only.

myUserTypeNB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance

myUsrParaList

Privilege: Private
Data Type: RWTValSlist<RWCString>
Default Value: NOT IDENTIFIED
No Inheritance
A list of PGE parameters for this production request.

myUsrParaValueList

Privilege: Private
Data Type: RWTValSlist<RWCString>
Default Value: NOT IDENTIFIED
No Inheritance
A list of parameter values for the PGE associated with this production request.

Operations:

`EctFloat CalculateResourceUsageNB (enum)`

Privilege: Public

No Inheritance

Determines the resource usage of a Production Request by multiplying the number of DPRs needed to satisfy this request with the `PIResourceRequirement` and `PIPerformance` values in `PIPGEProfile` for the PGE in the request.

`void DefinePGERuns ()`

Privilege: Public

No Inheritance

Defines the DPRs corresponding to the PGE.

`void Modify (RWCString:Modfield, RWCString:ModVal)`

Privilege: Public

No Inheritance

Modifies the production request.

`void PlProductionRequest ((RWCString:Product, RWCString:PGE, RWTime:Start, RWTime:Stop, ECTInt:Prior))`

Privilege: Public

No Inheritance

Constructor for the `PlProductionRequest` class.

`void RetrieveAllProdReq ()`

Privilege: Public

No Inheritance

Retrieves all the production requests currently in effect.

`void StoreProductionRequest (PlProductionRequestB)`

Privilege: Public

No Inheritance

Stores the production request in the PDPS database.

5.12.2.18 Class `PIResourceRequirement`

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: `PIPerformance(Public)`

Class: `PIGroundEvent(Public)` hasa

Class: `PIPGE(Public)` isreferencedby

Class: `PIDPRB(Public)` isrequiredby

`PIPGEProfile` (Aggregation)

Description:

This class contains a description of the resource requirements of a PGE, which may be matched against the resource configuration known to the Planning subsystem.

Attributes:

myComputer

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

A computer within the string required by the PGE

myDiskSpace

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

The disk space required for a PGE

myNCPUs

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

The number of CPUs required for a PGE

myOperatingSystem

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

The operating system for which a PGE is configured

myString

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

The string required by the PGE

Operations:

```
PlResourceRequirement PlResourceRequirement ()
```

Privilege: Public
No Inheritance
Constructor method

```
void ~PlResourceRequirement ()
```

Privilege: Public
No Inheritance
Destructor method

5.12.2.19 Class PIRoutineArrival**Synopsis:**

Parent Class: PIDataSource
Is Not A Distributed Object
Is Associated With:
This class is derived from the class PIDataSource

Description:

This class is a specialization of the PIDataSource class and describes the most frequent method for predicting data arrivals within the ECS (at least for the TRMM data sets). This class contains the attributes and operations required to describe routine ingest of external data.

Attributes:

```
myBoundary
```

Privilege: Private
Data Type: PIBoundary
Default Value: NOT IDENTIFIED
No Inheritance
A definitive boundary time from which predicted times can be derived.

```
myDelay
```

Privilege: Private
Data Type: EcTULongInt
Default Value: NOT IDENTIFIED
No Inheritance
The average delay between data collection and the arrival that of the Data Granule within ECS specified in seconds.

myPeriod

Privilege: Private

Data Type: PPeriod

Default Value: NOT IDENTIFIED

No Inheritance

A period of time (hour, day, week, set of orbits) used in predicting arrivals.

myPredictedMethod

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: PDataSource

Describes the method by which the data availability prediction occurs - e.g., routine arrival, arrival at scheduled times, FOS based prediction.

mySupplierName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PDataSource

Identifies the supplier of the data.

Operations:

void PlRoutineArrival ()

Privilege: Public

No Inheritance

void PlRoutineArrival (const RWCString &)

Privilege: Public

No Inheritance

Constructor method that is created from the database.

EcTVoid PredictArrivals (const PlTime &, const PlTime &)

Privilege: Public

No Inheritance

Method for predicting the arrival of data granules within an interval.

void ~PlRoutineArrival ()

Privilege: Public

No Inheritance

void PlDataSource ()

Privilege: Public
Inherited From: PlDataSource
default constructor

void PlDataSource (const RWCString &)

Privilege: Public
Inherited From: PlDataSource
overloaded constructor from data type id

EctVoid PredictArrivals (const PlTime &, const PlTime &)

Privilege: Public
Inherited From: PlDataSource
Predicts data instance arrivals for a given period, pure virtual operation.

void ~PlDataSource ()

Privilege: Public
Inherited From: PlDataSource
destructor method

5.12.2.20 Class PITile

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PlCluster(Public)

Description:

This defines a tile used by a TileScheduled PGE. It is a geographic area that will be processed by the PGE.

Attributes:

myCoordinates

Privilege: Private
Data Type: List of EctFloat
Default Value: NOT IDENTIFIED
No Inheritance
These are the coordinates of the tile, the four Latitude/Longitude pairs that define the area covered by the tile.

myTileID

Privilege: Private

Data Type: EctInt

Default Value: NOT IDENTIFIED

No Inheritance

This is the identifier for the tile. Each tile in a cluster will have a unique identifier.

Operations:

void PlTile ()

Privilege: Public

No Inheritance

This is the default constructor for the class.

void PlTile (EctInt TileID)

Privilege: Public

No Inheritance

This constructor takes in a TileID when it creates an instance of PlTile.

List of EctFloat ReturnCoordinates ()

Privilege: Public

No Inheritance

This method returns the coordinates of the tile stored in myCoordinates.

EctInt ReturnTileID ()

Privilege: Public

No Inheritance

This method returns the value in myTileID.

void SetCoordinates (List of EctFloat Coordinates)

Privilege: Public

No Inheritance

This method sets the coordinates for the tile.

void ~PlTile ()

Privilege: Public

No Inheritance

This is the destructor for the class.

5.12.2.21 Class PITileScheduledNB

Synopsis:

Parent Class: PIPGE
Is Not A Distributed Object
Is Associated With:
Class: PICluster(Public)
Class: PISourcetoDsHistoryNB(Private) references

Description:

This class is a table used by the PGEs scheduled for tiling to generate the DPRs and their input data.

Attributes:

myBoundary

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
A time from which all periods are referenced. For tile clusters, it is the time of the first orbit of a cluster.

myDelayFactor

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
No Inheritance
An amount of time used in calculating cluster availability which accounts for deviations in predictions.

myL0DataType

Privilege: Private
Data Type: ESDT
Default Value: NOT IDENTIFIED
No Inheritance
The LO data type that the tiles will be based upon. Used to calculate the time to move the L0 data from EDOS to the dataserver.

myNumClusters

Privilege: Private
Data Type: EcTInt
Default Value: NOT IDENTIFIED
No Inheritance
The number of clusters for this PGE.

myPeriod

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

The amount of time for a complete set of orbits that include a cluster of tiles.

myProductionDeltaTime

Privilege: Private

Data Type: RWTime

Default Value: NOT IDENTIFIED

No Inheritance

Estimated time to produce all intermediate products used as input to produce tiles.

myTileDefinition

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is used to create the tile or spatial based input data.

myDefaultTimer

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

Inherited From: PIPGE

This is the default timer value for the PGE for use in Alternative Input selection.

myInputDataTypeList

Privilege: Private

Data Type: RWTValslist<RWCString>

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

List of input data types needed by the PGE

myInstrument

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

Instrument for which the PGE is appropriate

myNumCPUs

Privilege: Private

Data Type: EcTInt

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

This is the number of CPUs that the PGE requires. It should be 1 for non-parallel processing PGEs.

myOutputDataTypeList

Privilege: Private

Data Type: RWTValSlist<RWCString>

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

List of output data types needed by the PGE

myPGENAME

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

Name of the PGE

myPGEVersion

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

Version number of the PGE

myPlatform

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

Platform for which the PGE is appropriate, may be a list

myTestOperational

Privilege: Private

Data Type: enum

Default Value: NOT IDENTIFIED

Inherited From: PIPGE

Indicates if the PGE's status is test or operational

myPgeId

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIPGE
This is the unique identifier of the PGE.

myType

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: PIPGE
This is the type of the PGE.

myNormalStagingTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
Inherited From: PIPGE
This is the time to stage the input data (during normal processing) for this PGE.

myReprocessingStagingTime

Privilege: Private
Data Type: RWTime
Default Value: NOT IDENTIFIED
Inherited From: PIPGE
This is the time to stage the input data of the PGE when performing Reprocessing.

Operations:

void CalcTileOverlap ()

Privilege: Public
No Inheritance
This calculates any geographic or input data overlap between tiles.

void GenerateDPRs (PlProductionRequest)

Privilege: Public
No Inheritance
Generates a DPR for each tile in each cluster.

`void PlTileScheduledNB ()`

Privilege: Public

No Inheritance

This is the default constructor for the class.

`PlTileScheduledNB (ESDT L0Type, RWTime PredictionDelta,
RWTime ProductionDelta, RWCString Definition)`

Privilege: Protection Not Identified

No Inheritance

This constructor takes in L0Type, PredictionDelta, ProductionDelta, and Definition and creates an object of the class, giving the corresponding attributes those values.

`ESDT ReturnL0Type ()`

Privilege: Public

No Inheritance

Returns myL0DataType.

`RWTime ReturnProductionDelta ()`

Privilege: Public

No Inheritance

Returns myProductionDeltaTime.

`void ~PlTileScheduledNB ()`

Privilege: Public

No Inheritance

This is the destructor for the class.

`void Delete ()`

Privilege: Public

Inherited From: PIPGE

Delete the PGE from the PDPS database

`void FindDataAvailability (Interval)`

Privilege: Public

Inherited From: PIPGE

This is used to find what input data is available for this PGE.

`void GenerateDPRs (PlProductionRequest)`

Privilege: Public

Inherited From: PIPGE

Generate the Data Processing Requests to fulfill the Production Request.

`void Modify (TestOrOper:enum)`

Privilege: Public

Inherited From: PIPGE

This is used to modify the attributes of this class.

`PlPGE PlPGE ()`

Privilege: Public

Inherited From: PIPGE

Constructor method

`PlPGE (int PGEid)`

Privilege: Protection Not Identified

Inherited From: PIPGE

Construct the PGE from the PDPS database

`void UpdateVersion ()`

Privilege: Public

Inherited From: PIPGE

Update the version of the PGE

`void ~PlPGE ()`

Privilege: Public

Inherited From: PIPGE

Destructor method

`PlPGE PlPGE (RWCString:PGEid)`

Privilege: Public

Inherited From: PIPGE

This constructor takes the PGE ID of the object to be constructed.

5.12.2.22 Class PIUserParameters

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: PIPGE(Public) isreferencedby

PIPGEProfile (Aggregation)

Description:

Describes any user defined parameters that are associated to a PGE.

Attributes:

myDefaultValue

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

The default value for the user parameter.

myDescription

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

Describes the user parameter

myLogicalID

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

Id of the user parameter

myName

Privilege: Private

Data Type: String

Default Value: NOT IDENTIFIED

No Inheritance

Name of the user parameter

Operations:

void Create ()

Privilege: Public

No Inheritance

Create an entry within the PDPS database

void Delete ()

Privilege: Public

No Inheritance

Delete the entry from the PDPS database

```
void Modify (DefValue:String)
    Privilege: Public
    No Inheritance
    Modifies the default value of the user parameter
```

```
PlUserParameters PlUserParameters ()
    Privilege: Public
    No Inheritance
    Constructor method
```

```
void ~PlUserParameters ()
    Privilege: Public
    No Inheritance
    Destructor method
```

5.13 Processing Classes

5.13.1 Processing Classes Overview

PRONG - Processing CSCI - The Processing CSCI provides the services required to manage and monitor the Science Data Processing environment which is used to generate data products using Science Software (PGEs) provided by the instrument teams.

5.13.2 Processing Class Descriptions

5.13.2.1 Class DpPrScheduler

Synopsis:

```
No Parent Class
Distributed Object
Is Associated With:
Class: DpPrDprStatusNB(Private) Creates
Class: PlPerformance(Private) GetStagingTimeFrom
Class: DpPrDataManagement(Private) Initializes
Class: DpPrCotsManager(Private) ManageJobs
Class: PlDPR(Public) ManagesDPRJobs
Class: PlGroundEvent(Private) ManagesGroundEvents
Class: PlPge(Private) ObtainsInformationAboutRunConditionsFrom
```

Description:

DpPrScheduler provides operations to manage science software on a DPR level.

Attributes:

Operations:

`void CancelDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

This operation cancels the Job Box and all jobs associated with the input DPR. The Data Manager is contacted to release all resources reserved for the DPR, and to update its data as to the number of DPRs requiring a given data file.

`void CancelGEvntJob (Event:PlGroundEvent)`

Privilege: Public

No Inheritance

This operation is used by Planning to cancel a previously scheduled Ground Event.

`void CreateDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

This operation is used to convert a Data Processing Request (DPR) into a series of "jobs" to be processed by the Scheduling COTS package. Planning creates a DPR and passes it to the DpPrScheduler via this operation. The information in the DPR is used to create a "Job Box", containing all the steps necessary to successfully run the PGE associated with the DPR. Individual jobs are created for setup of execution resources and for ensuring that all necessary input data is local, as well as running the PGE itself and deallocating the resources and data requirements associated with the DPR. These jobs are entered into the Scheduling COTS software (via the DpPrCotsManager class) to be started when a) All PGE Dependencies have been satisfied, and b) All external data is available at the Data Server.

`void CreateGEvntJob (Event:PlGroundEvent)`

Privilege: Public

No Inheritance

This operation allows Planning to schedule Ground Events in the daily production schedule. Ground Events describe the allocation of a resource to a non-production task such as maintainance.

`DpPrProcessingStatus GetDprJobStatus (Dpr:PlDpr)`

Privilege: Public

No Inheritance

This operation returns Processing Status in the Scheduling COTS of the Job Box associated with the DPR. Values can include ON_HOLD, STARTING, WAITING, RUNNING, SUCCESS or FAILURE.

`void ReleaseDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

This operation is used by Planning to inform the Scheduler that all input data required from the Data Server is available. At this time, the Scheduler releases the Job Box via the CotsManager, which allows the jobs associated with the DPR to start processing when all the PGE Dependencies have been satisfied (i.e., if all the PGE Dependencies are satisfied but Planning has not called this operation, the DPR cannot be processed).

`Status ResumeDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

`Status SuspendDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

`void UpdateDprJob (Dpr:PlDpr)`

Privilege: Public

No Inheritance

This operation is used by Planning to modify a DPR's Priority or Time Information only. Time Information includes minimum and/or maximum start times, end times, and total predicted processing times. These are used for error detection (i.e., if a DPR is taking much longer than predicted to run, an Alarm can be raised). Planning modifies the DPR and passes it to the Scheduler, which contacts the Scheduling COTS (through the CotsManager) to update the information.

5.13.2.2 Class GIUR

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: DpPrQaMonitor(Private) GetDataUsing

Class: PlDataType(Private)

Description:

This is the abstract base class for all Universal Reference (UR)s. A UR is a special ECS identifier for an object. What makes it special is that an object can be identified, but the object does not have to exist in memory at the time. The contents of a UR are specified by subclasses.

Generally speaking, the contents are the key elements of the object that this UR refers to. It can be thought of as DNA. We can reconstitute or clone an organism (i.e. object or URProvider) given its DNA (i.e. UR). The key public methods are "Externalize" and "Internalize"

Attributes:

Operations:

5.13.2.3 Class PIDPR

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: DpPrPge(Private) Locates
Class: DpPrScheduler(Public) ManagesDPRJobs
Class: DpPrDprStatusNB(Private) ReportsStatus
Class: DpPrDataManager(Private) locates
Class: PIDataGranule(Private) specifies

Description:

This class describes an individual run of a PGE.

Attributes:

myActualStart
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myCompletionState
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myDprid
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myInputDataInstanceList List
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myOutputDataInstanceList List
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myPredictedStart
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

myPriority
Privilege: Private
Default Value: NOT IDENTIFIED
No Inheritance

Operations:

void Cancel ()
Privilege: Public
No Inheritance

void CheckAvailability ()
Privilege: Public
No Inheritance

void GetCommandString ()
Privilege: Public
No Inheritance

void GetInputDataInstance ()
Privilege: Public
No Inheritance

GetNextInputData ()
Privilege: Protection Not Identified
No Inheritance

void GetOutputDataInstance ()
Privilege: Public
No Inheritance

```

void Modify ()
    Privilege: Public
    No Inheritance

PlDPB ()
    Privilege: Protection Not Identified
    No Inheritance

void PlDPR ()
    Privilege: Public
    No Inheritance

void Release ()
    Privilege: Public
    No Inheritance

void Schedule ()
    Privilege: Public
    No Inheritance

void Status ()
    Privilege: Public
    No Inheritance

~PlDPB ()
    Privilege: Protection Not Identified
    No Inheritance

void ~PlDPR ()
    Privilege: Public
    No Inheritance

```

5.13.2.4 Class PIDataTypeB

Synopsis:

```

No Parent Class
Is Not A Distributed Object
Is Associated With:
Class: PIPGE(Private)
Class: DpPrDatabaseValNB(Private) identifies
Class: PIDataGranule(Private) matches

```


Description:

This class describes a data type known to the planning subsystem. This is a description of an input or output type, distinct to a granule or instance of the data type. The class is an abstraction or proxy that describes one of the Data Server ESDTs. The class captures data and operations that are required to subscribe and receive notification from the Data Server when a new instance of the Data Type arrives.

Attributes:**Operations:**

5.14 Science Data Server Classes

5.14.1 Science Data Server Classes Overview

The Science Data Server CSCI (SDSRV CSCI) expresses the "data are data" concept of the architecture. By acting as the session manager and primary client interface to the data, the SDSRV provides access to the data server holdings using methods and interfaces which remain consistent regardless of how the data is stored. Access (searching, requesting) and instantiation (creation, schema) of the data server data holdings are represented as Earth Science Data Type classes (ESDTs). The Computer Science Data Type (CSDT) classes represent the physical organization and storage implementation of the ESDTs. ESDT and CSDT are implemented by combining off the shelf DBMS technology with software developed to support the aspects that are unique to the ECS data and are not supported by the off the shelf software directly. For example, the DBMS (perhaps with third party software extensions which are also off-the-shelf) will support data types such as integer, floating point, string, arrays, images, time, and various kinds of spatial objects. However, when building database support for ECS products, software may need to be developed to support coordinate transformations which are not supported by the off-the-shelf software. Information which resides in a DBMS is defined in a formal fashion. The definition is called a database schema. The DBMS uses it to interpret the queries and access requests it receives and execute them. If all support needed by the ECS science data were available off-the-shelf within the DBMS, the SDSRV software model would be very simple. It would consist, for the most part, of the DBMS. The data objects, i.e., the ESDT, are defined in the schema and are not visible in the software design of the SDSRV itself. The schema would implement the requirements of the ECS Core Meta Data Model (with product unique extensions). The Core Meta Data Model would not be reflected in the software design, only in the database schema. In actual fact, the SDSRV needs to add software support for science data types, for example, when manipulating if for distribution, or to implement browse, subsetting, and subsampling. As a result, the SDSRV software design is a hybrid of some of the types in the ECS Core Meta Data Model, and other software object classes which are needed to implement other aspects of the data server. An example of this is the Session class. It provides clients with a uniform, stateful interface with the data server (rollback and suspension, operations on working collections), independent of the underlying servers for the ESDTs. The application of inheritance, polymorphism, and specialization constructs into the design enables the extensibility of the data server to evolve the structure and behavior of existing data holdings, as well as to add new ones.

5.14.2 Science Data Server Class Descriptions

5.14.2.1 Class DsClAction

Synopsis:

Parent Class: DsSbActionBase
Parent Class: DsShAction
Is Not A Distributed Object
Is Associated With:
Class: DsClSubscription(Public) actionof

Description:

A client interface object that represents the components of the action to be performed when a subscription is triggered. The possibilities are that the client will receive a notification (including all parameters that are returned by the object that triggers the subscription and an optional piece of client-specified text) and/or a request that will be executed. The client is required to specify an action for each subscription.

Attributes:

myRequest

Privilege: Private
Data Type: DsClRequest
Default Value: NOT IDENTIFIED
No Inheritance
The request that is currently associated with this action.

myNotifyFlag

Privilege: Private
Data Type: RWBoolean
Default Value: NOT IDENTIFIED
Inherited From: DsSbActionBase
Flag which determines if this action is a notification.

myRequestFlagB

Privilege: Private
Data Type: RWBoolean
Default Value: NOT IDENTIFIED
Inherited From: DsSbActionBase
Flag which determines if this action is a request.

myText

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
Inherited From: DsSbActionBase
Defines text of an action.

Operations:

void ClearRequestB ()

Privilege: Public
No Inheritance
Used to clear any request that has been set for the action.

void DsClAction ()

Privilege: Public
No Inheritance
Constructs a default action.

void DsClActionB (RWCString &text, DsClRequest * = NULL)

Privilege: Public
No Inheritance
Used to construct an action from a piece of text and, RELEASE B: optionally, a request. The notification flag is set.

void DsClActionB (DsClRequest &, RWBoolean = FALSE, RWCString * = NULL)

Privilege: Public
No Inheritance
Used to construct an action from a request and, optionally, a value for the notification flag and a piece of text.

const DsClRequest& GetRequestB ()

Privilege: Public
No Inheritance
Returns the request currently set for the action.

void SetRequestB (DsClRequest &)

Privilege: Public
No Inheritance
Sets the request to be executed when the subscription fires.

```

void ~DsClAction ()
    Privilege: Public
    No Inheritance
    Used to destroy an action.

void ClearRequestB ()
    Privilege: Public
    Inherited From: DsSbActionBase
    Provides the action base class the ability to have requests cleared.

void DsSbActionBase (RWBoolean notify = FALSE, const
RWCString *text = 0)
    Privilege: Public
    Inherited From: DsSbActionBase
    Constructs an Action base class setting the notification flag and text
    attribute.

void DsSbActionBase ()
    Privilege: Public
    Inherited From: DsSbActionBase
    Constructs a DsSbActionBase.

RWBoolean GetNotify ()
    Privilege: Public
    Inherited From: DsSbActionBase
    Accesses the notification flag.

const RWCString & GetText ()
    Privilege: Public
    Inherited From: DsSbActionBase
    Accesses the text attribute.

RWBoolean HasRequestB ()
    Privilege: Public
    Inherited From: DsSbActionBase
    Checks whether the action contains requests.

void SetNotify (RWBoolean)
    Privilege: Public
    Inherited From: DsSbActionBase
    Sets notification flag.

```

```
void SetText (RWCString &)
```

Privilege: Public
Inherited From: DsSbActionBase
Sets the text attribute.

```
void ~DsSbActionBase ()
```

Privilege: Public
Inherited From: DsSbActionBase
Removes an object of this type.

5.14.2.2 Class DsClCommand

Synopsis:

Parent Class: DsSrCommandBase
Is Not A Distributed Object
Is Associated With:
DsClRequest (Aggregation)

Description:

A specialization of the DsCommand class for client interfaces. Adds constructors that ease building of commands based on advertisements, or special direct commands that are "built-in" to the data server and do not correspond to advertisements.

Attributes:

```
myInfo
```

Privilege:
Data Type: DsSrCommandInfo*
Default Value: NOT IDENTIFIED
Inherited From: DsSrCommandBase
A pointer to the core command information for this command. (Which is, In OO terms, the implementation for this interface).

Operations:

```
void DsSrCommandBase ()
```

Privilege: Public
Inherited From: DsSrCommandBase
Used to construct an empty command.

```
void DsSrCommandBase (svc: RWCString &, pl: GlParameterList  
*, DsESrCommandCategory)
```

Privilege: Public
Inherited From: DsSrCommandBase
Used to construct a command with the given service, parameters, and category.

DsESrRequestCategory GetCategory ()

Privilege: Public

Inherited From: DsSrCommandBase

Used to retrieve the current category for this command.

DsSrCommandInfo* GetInfo ()

Privilege: Public

Inherited From: DsSrCommandBase

Returns a pointer to the underlying data (DsSrRequestInfo) for this request.

const GlParameterList & GetParameters ()

Privilege: Public

Inherited From: DsSrCommandBase

Returns the parameters for this command.

const RWCString & GetServiceName ()

Privilege: Public

Inherited From: DsSrCommandBase

Returns the current service name for this command.

void SetCategory (DsESrRequestCategory)

Privilege: Public

Inherited From: DsSrCommandBase

Used to set the category for the command.

void SetParameters (pl: GlParameterList &)

Privilege: Public

Inherited From: DsSrCommandBase

Used to set the parameters for this command.

void SetServiceName (svc: const RWCString &)

Privilege: Public

Inherited From: DsSrCommandBase

Used to set the service name for this command.

void ~DsSrCommandBase ()

Privilege: Public

Inherited From: DsSrCommandBase

Used to destroy a command.

`void DsClCommand (SpecialCommand)`

Privilege: Public

No Inheritance

Used to construct "special" commands, i.e. commands that are not advertised. For example, the command to reset the working collection.

`void DsClCommand (adv: Advertisement &, parms:
GlParameterList &)`

Privilege: Public

No Inheritance

Used to construct a command from an advertisement. The GlParameterList gives the parameters required by the chosen command.

`void DsClCommand ()`

Privilege: Public

No Inheritance

Constructs an empty command.

`void DsClCommand (svc: RWCString&, pl: GlParameterList&, cat:
DsESrCommandCategory)`

Privilege: Public

No Inheritance

Used to construct a command from its basic parts: service name, parameters, and category.

`RWCString Textify ()`

Privilege: Public

No Inheritance

Used to convert a command into a human-readable format.

`void ~DsClCommand ()`

Privilege: Public

No Inheritance

Used to destroy a command.

5.14.2.3 Class DsClDescriptor

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

None

Description:

This object provides access to services pertaining to the definition of a specific data type. This includes access to metadata configuration information and queriable parameters. This object validates metadata.

Attributes:

`myStatus`

Privilege: Private

Data Type: `GlStatus`

Default Value: NOT IDENTIFIED

No Inheritance

The status attribute captures status information about activities that are performed on this instance.

`myTypeID`

Privilege: Private

Data Type: `DsGeTypeID *`

Default Value: NOT IDENTIFIED

No Inheritance

Reference to an object that uniquely identifies the specific type of this descriptor.

`ourCollectorVector`

Privilege: Private

Data Type: `RWVector *`

Default Value: NOT IDENTIFIED

No Inheritance

Reference to a static vector of collectors. Each collector is connected to a data server. Upon construction, the descriptor must check to see if there is a collector connected to its data server. If so, the descriptor add itself to that collectors list of items. If not, the descriptor creates a collector that talks to the desired data server and adds it to the static vector collector set.

Operations:

`void DsClDescriptor (GlClient &, GlUR &,DsGeTypeID &)`

Privilege: Private

No Inheritance

Constructor for the descriptor. The arguments are references to a client object, a dataserver, and a `TypeID` object indicating the type of the descriptor to create.

DsClDescriptorCollector* GetCollector ()

Privilege: Private

No Inheritance

Returns a pointer to the DescriptorCollector that contains this descriptor.

GlStatus GetMCF (ostream &)

Privilege: Public

No Inheritance

Returns a stream containing the metadata configuration file (MCF) data.

RWBoolean GetQueryableParameters (GlParameterList &)

Privilege: Public

No Inheritance

The service takes a reference to an empty GlParameterList and fills that list in with the queryable parameters and their associated types for this data type.

DsGetTypeID & GetTypeID ()

Privilege: Public

No Inheritance

Returns a reference to the DsGetTypeID object which contains information to uniquely identify the type of this descriptor.

void SetTypeID (DsGetTypeID &)

Privilege: Public

No Inheritance

This is a private function to set the typeID for this object.

GlStatus ValidateB (istream &metadata)

Privilege: Public

No Inheritance

This service takes a stream which contains metadata and validates this metadata. The GlStatus object that is returned provides an indication of whether the validation was successful.

void ~DsClDescriptor ()

Privilege: Public

No Inheritance

Destructor for the descriptor. If this is the last descriptor of the hidden collector to be destroyed, then the collector is also destroyed.

5.14.2.4 Class DsCIESDTRreference

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
DsCIESDTRreferenceCollector (Aggregation)
DsCIESDTRreferenceVector (Aggregation)

Description:

This object is a reference to an ESDT that is within a DataServer's holdings. This object provides services that are homogeneous for all ESDTs.

Attributes:

myCollector

Privilege: Private
Data Type: DsCIESDTRreferenceCollector *
Default Value: NULL
No Inheritance

A pointer to the collector that this reference is a member of. If this pointer is null, then this reference is a member of one of the collectors in the static collector vector.

myCreateDate

Privilege: Private
Data Type: RWDate
Default Value: NOT IDENTIFIED
No Inheritance

The creation date of the ESDT that this reference represents.

myReferenceVector

Privilege: Private
Data Type: DsCIESDTRreferenceVector *
Default Value: NOT IDENTIFIED
No Inheritance

The DsCIESDTRreferenceVector that this DsCIESDTRreference belongs to.

mySize

Privilege: Private
Data Type: size_t
Default Value: NOT IDENTIFIED
No Inheritance

This number of MB that the underlying granule would occupy if the ESDT that this reference points to was externalized.

myStatus

Privilege: Private
Data Type: GlStatus
Default Value: NOT IDENTIFIED
No Inheritance
Current status of the ESDT Reference object.

myTypeInfo

Privilege: Private
Data Type: DsClTypeInfo *
Default Value: NOT IDENTIFIED
No Inheritance
A pointer to an object which contains information related to all ESDT references of the same type.

myUR

Privilege: Private
Data Type: UR
Default Value: NOT IDENTIFIED
No Inheritance
The unique reference that is assigned to this ESDT reference.

ourCollectorVector

Privilege: Private
Data Type: DsClCollectorVector<DsClESDTRreferenceCollector>
Default Value: NOT IDENTIFIED
No Inheritance
A static vector of ESDT collectors. Each ESDT reference is collected by an ESDT collector. The collector vector is used to allow creation of an ESDT reference without first establishing a collector.

Operations:

```
void DsClESDTRreference (theDSS:GlUR &,  
theBoss:DsClESDTRreferenceCollector *= NULL)
```

Privilege: Public
No Inheritance
This constructor takes as arguments the UR of the Data Server, a reference to the Reference collector that this ESDT Reference should be made a member of.

`const DsClESDTRedirectCollector & GetCollector ()`

Privilege: Public

No Inheritance

Returns a reference to the DsClESDTRedirectCollector that contains this ESDTRedirect.

`const RWDat & GetCreateDate ()`

Privilege: Public

No Inheritance

Returns the date that the underlying ESDT granule was created.

`const GlStatus & GetQueryableParameters (GlParameterList &)`

Privilege: Public

No Inheritance

This service is used to obtain the queryable parameters for this ESDT type. The queryable parameters will be the same for all ESDT references of the same type. For example, all CER03s have the same queryable parameters. The input argument is a reference to an empty parameter list. The return value is a boolean indicating success or failure of the request. If the request was successful, the parameter list will have the names and types (but no values) of all queryable parameters for this TYPE.

`DsClESDTRedirectVector * GetReferenceVector ()`

Privilege: Private

No Inheritance

This private operation returns the pointer to this object's DsClESDTRedirectVector.

`const GlStatus & GetServiceAppletB (theSvc:RWCString &,
svcUR:GlUR&, theVersion:RWCString &=NULL)`

Privilege: Public

No Inheritance

This operation will request the GUI applet which allows the user to view the data produced by a given service.

`const GlStatus & GetServiceConfigurationB (theSvc:RWCString
&, theConfig:GlParameterList &)`

Privilege: Public

No Inheritance

This operation allows the user to get the necessary configuration information for the service applet on the client desktop.

`size_t GetSize ()`

Privilege: Public

No Inheritance

Returns the size of data granule associated with this ESDT Reference. This is the size of the granule if externalized.

`const DsGetTypeID & GetTypeID ()`

Privilege: Public

No Inheritance

Returns a reference to an object that contains all of the information that uniquely identifies this type.

`const DsClTypeInfo & GetTypeInfo ()`

Privilege: Private

No Inheritance

Returns a reference to the `DsClTypeInfo` object for this ESDT Reference. This `DsClTypeInfo` object contains information common to all ESDT References of the same type (ESDT).

`const GlUR & GetUR ()`

Privilege: Public

No Inheritance

Returns the UR that is associated with this ESDT reference.

`const GlStatus & Inspect (GlParameterList &)`

Privilege: Public

No Inheritance

This service returns values for given parameters. The parameter list has the names of the parameters for which values are desired and this service returns the same parameter list with the values and associated types filled in. A boolean is also returned to indicate success or failure of the service request.

`void SetCollector (DsClESDTRreferenceCollector *)`

Privilege: Private

No Inheritance

This is a private service used internally to set the `ESDTRreferenceCollector` attribute.

```
void SetCreateDate (RWDate &)
```

Privilege: Private

No Inheritance

This is a private service used to set the myCreateDate attribute.

```
void SetReferenceVector (DsClESDTReferenceVector *)
```

Privilege: Private

No Inheritance

This is a private service used to set the myReferenceVector attribute.

```
void SetSize (size_t)
```

Privilege: Private

No Inheritance

This is a private member function used to set the mySize attribute.

```
void SetTypeInfo (DsClTypeInfo *)
```

Privilege: Private

No Inheritance

This is a private service used to set the myTypeInfo attribute.

```
void SetUR (GLUR &)
```

Privilege: Private

No Inheritance

This is a private service used to set the myUR attribute.

```
void SetVersionB (RWCString &)
```

Privilege: Private

No Inheritance

This operation allows the user to specify which version of a product is being requested.

```
const GLStatus & Submit (DsClRequest &)
```

Privilege: Public

No Inheritance

This public operation is a convenience function to allow the client programmer to submit requests to this object. This operation just calls the Submit method on the provided DsClRequest.

```
void ~DsClESDTRedference ( )
```

Privilege: Public

No Inheritance

Destructor for the ESDT Reference. If this descriptor is a member of a collector in the static collector vector and this is the last descriptor to be destroyed, then its collector is also destroyed.

5.14.2.5 Class DsClESDTRedferenceCollector

Synopsis:

Parent Class: DsClCollector

Parent Class: DsClGenSuspendor

Distributed Object

Is Associated With:

This class is derived from the class DsClGenSuspendor

Description:

This public, distributed class is a specialization of the Collector class which handles DsClESDTRedferences. This class is much more complex than the base class. This class provides, in addition to the normal set operations for ESDTRedferences, the ability to handle requests, working-collection synchronization, and sessions. It also contains private operations to hand the ESDTRedference-level actions to the dataserver.

Attributes:

mySearchCallback

Privilege: Private

Data Type: GICallback

Default Value: NOT IDENTIFIED

No Inheritance

This attribute is for specifying the local callback for a query.

myStateB

Privilege: Private

Data Type: DsECIState

Default Value: Active

No Inheritance

This attribute records the current state of the object. This is used to control whether or not the DsClESDTRedferenceCollector will accept inputs or not (i.e., when it is in "suspended" state, no inputs from client software are allowed).

myStatus

Privilege: Private

Data Type: GlStatus

Default Value: NOT IDENTIFIED

No Inheritance

This attribute allows the object to maintain information on current status.

Operations:

void AddESDTRreferenceB (const DsClESDTRreference *ERef)

Privilege: Public

No Inheritance

This operation is used to instantiate a DsClESDTRreference when the (empty) object has already been allocated. For example, if the client creates a DsClESDTRreference with a UR (which creates a DsClESDTRreferenceCollector), this will turn into a request for the given UR. However, once the ESDT has been found and instantiated on the server side, and UpdateState will make the client side collection "match", the DsClESDTRreference object that the client created must be used to store the reference, as the client software only knows about that actual memory address. So the UpdateState method will use a "replace" command to "fill in" the existing object which the client software has created. This is also useful for subsetting, when a DsClESDTRreference already exists, but the metadata which describes it may change (become a smaller set) as a result of the subsetting operation. The same DsClESDTRreference object should still be used, even though its contents have changed.

const GlStatus & BuildRequestVectorB (GlURVector &)

Privilege: Protected Operation

No Inheritance

This private operation allows the DsClESDTRreferenceCollector object to reconstruct the state of a session in the case that a user has "logged off" during a long-running request, and reconnects to the session to find the results. This operation reestablishes the client-side request structure (DsClSubmittedRequests) to check the status of the actual requests on the server side.

DsClESDTRreference* CreateESDTRreference (GlUR &)

Privilege: Protected Operation

No Inheritance

This operation actually creates a DsClESDTRreference object. It is used by the UpdateState method when an "add" command is received (i.e., no DsClESDTRreference currently exists for the given UR).

`RWBoolean DeleteESDTRreference (DsClESDTRreference*)`

Privilege: Protected Operation

No Inheritance

This operation is used during in the UpdateState method to actually remove client DsClESDTRreference objects when their corresponding server-side objects have been removed from the DsSrWorkingCollection.

`void DsClESDTRreferenceCollector (GlUR &dataserver,
MSS_UserProfile &, DsTSessionID = NULL)`

Privilege: Public

No Inheritance

This constructor is used by a data server client to establish a session with the specified science data server. The client application supplies a reference to the desired science data server as well as the user's MSS profile.

`void DsClESDTRreferenceCollectorB (MSS_UserProfile &, GlUR
&session)`

Privilege: Public

No Inheritance

This version of the constructor allows the user to supply the UR of a previously suspended session, in effect resuming the session.

`const DsClRequestVector & GetRequestVector ()`

Privilege: Public

No Inheritance

This operation allows client software to get a reference to the set of requests submitted during the current session, for whatever local processing (iteration) desired.

`const GlStatus & GetRequestsB (GlURVector &)`

Privilege: Protected Operation

No Inheritance

This private operation allows the DsClESDTRreferenceCollector object to get a list of all the requests which have been submitted for the current session. This feature is to support the situation where a user has logged out while a long request is processing, then makes a new connection to the same SessionID to get the results. The collector object will use this operation in the "rebuilding" of the session state.

`const GlStatus & GetSessionLogB (ostream &outfile)`

Privilege: Public

No Inheritance

This operation allows the client software to declare an output stream and have the dataserver send over the portion of the system log which pertains to this particular session. All file-related programming is expected to occur in the client software.

`DsEC1State GetStateB ()`

Privilege: Public

No Inheritance

This operation provides access to the current state attribute.

`const GlStatus & GetStatusB ()`

Privilege: Public

No Inheritance

This operation allows the user to check the status of the DsCIESDTRreferenceCollector.

`GlURVector & ListURsB ()`

Privilege: Public

No Inheritance

This operation allows the client to get a list of the URs for all granules which are currently in the working collection.

`const GlStatus & RemoveESDTRreferenceB (DsC1ESDTRreference &)`

Privilege: Public

No Inheritance

This operation is used to allow a user to "remove" a DsCIESDTRreference object from a collection and save it elsewhere. Essentially, it will package the UR for the given DsCIESDTRreference so that it can be stored elsewhere and re-instantiated at a later time.

`const GlStatus & Reset ()`

Privilege: Public

No Inheritance

This operation removes all current state for the collection, i.e. removes all existing ESDTRreferences.

`const GlStatus & ResumeSessionB (GlUR &SuspendedSession)`

Privilege: Protected Operation

No Inheritance

This private operation supports the resumption of a previously suspended session. A session is "resumed" by constructing the DsCIESDTRreferenceCollector with a session UR. This supporting operation is used to indicate to the (rebuilt) server-side session that the client-side has been reconstructed, and the state flags can now be set from "suspended" to "active".

`const GlStatus & Search (DsClQuery &)`

Privilege: Public

No Inheritance

This operation takes a DsClQuery object, which has been defined and filled in by the client software, and creates a DsClRequest (and associated DsClSubmittedRequest) and submits the request to the dataserver. The return values are included in the DsClRequest object, as are whatever commands are necessary to update the client working collection (this collection) to match the working collection on the server. The DsCIESDTRreferenceCollector *callback* function, which gets called by DsClRequest upon completion, will take the results of the DsClRequest, invoke UpdateState to make the client-side working collection match the server side one, then get the DsClQuery pointer from the DsClRequest so that it can call the Query object's callback function.

`void SetDialogCallbackB (GlCallback&)`

Privilege: Public

No Inheritance

This operation allows the client software to declare a callback for handling dialog-type notifications. It is not expected to be used in Release A.

`void SetSearchCallback (GlCallback&)`

Privilege: Protected Operation

No Inheritance

This operation allows the client software to set a local callback for a query. This operation is private to the Search method.

`RWBoolean SetStateB (DsEClState)`

Privilege: Public

No Inheritance

This operation sets the current state.

`void SetStatusCallback (GlCallback&)`

Privilege: Public

No Inheritance

This operation allows the client software to establish a callback routine that will be the local callback function for all requests to report status to.

`const GlStatus & Submit (DsClRequest &)`

Privilege: Public

No Inheritance

This operation invokes the Submit of the DsClRequest object which establishes a DsClSubmittedRequest to handle the callback stuff. The DsClRequest will call the Collector (base class) SubmitToServer to actually send itself (request) across the network.

`const GlStatus & SuspendSessionB (GlUR &SuspendedSession)`

Privilege: Public

No Inheritance

This operation stores the state of the current session for later resumption. The UR of the session may be created at this point or it may be assigned when the session is first created.

`const GlStatus & UpdateState (GlParameterList &)`

Privilege: Protected Operation

No Inheritance

This private operation is used to make the client-space representation of the working collectino match that on the server side. It takes commands from the GlParameterList and executes them in turn. Because of the possibility of multiple simultaneous queries, this operation must be concurrency-safe. Commands take one of the following *logical* forms. remove(existing_UR) add(new_UR, type(parameters), date, size) replace(existing_UR, new_type(parameters), date, size) If the ESDTReferenceVector (by type) is retained, the logic is as follows. add: if type doesn't exist, create type insert new_UR remove: remove existing_UR if last UR in type, remove type replace: if new_type doesn't exist, create type set parameters of existing_UR to new values if last UR in old_type, remove old_type

`void ~DsClESDTReferenceCollector ()`

Privilege: Public

No Inheritance

The DsClESDTReferenceCollector's destructor.

```
void DsClCollector (GlUR &dataserver, MSS_UserProfile &)
```

Privilege: Private

Inherited From: DsClCollector

The default constructor for this class is private so that an object of this class can not be created without providing enough information to establish a connection (i.e., either a dataserver ID or a session ID).

```
DsESrConnectionID GetConnectionID ()
```

Privilege: Public

Inherited From: DsClCollector

This public operation allows client software to retrieve the dataserver-assigned session ID so that users can turn off their terminals during long-running request processing, then log on later and reconnect to the session.

```
const GlStatus & SubmitToServer (DsClRequest&)
```

Privilege: Protected Operation

Inherited From: DsClCollector

This protected operation passes a request from a DsClRequest to the dataserver after the DsClRequest has established the DsClSubmittedRequest to handle the distributed callback. This operation is fully implemented on the server side (i.e. in either DsSrConnection or the specialized connection object DsSrSession). All return values are handled by the DsClRequest object, which contains imbedded return parameters for this purpose.

```
void ~DsClCollector ()
```

Privilege: Public

Inherited From: DsClCollector

5.14.2.6 Class DsClNotificationReceiver

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

None

Description:

Handles asynchronous notifications from the data server to the currently logged-on client. Client will specify a callback point in his application code to receive these events (e.g. subscription triggering notifications) for each data server that s/he connects to.

Attributes:

`myCallback`

Privilege: Private

Data Type: `GICallback`

Default Value: NOT IDENTIFIED

No Inheritance

The callback that will be used when a notification is received.

`myReceiveFlag`

Privilege: Private

Data Type: `RWBoolean`

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether notifications should be passed to callback; initially FALSE.

Operations:

`void DsClNotificationReceiver (server: GLUR, callback: GICallback)`

Privilege: Public

No Inheritance

Used to construct a notification receiver for a given server. Notifications will be sent to the given callback.

`void DsClNotificationReceiver ()`

Privilege: Public

No Inheritance

Constructs a default notification receiver.

`GLStatus GetOneNotification ()`

Privilege: Protected Operation

No Inheritance

Used to wait for a notification to arrive. Starts a thread to wait, which can be stopped by `StopReceiving()`.

`void SetCallback (callback: GICallback)`

Privilege: Public

No Inheritance

Used to change the callback that is used when a notification arrives.

`void StartReceiving ()`

Privilege: Public

No Inheritance

Used by the client to control when notifications may be received. Must be called after creation to enable reception of notifications.

`void StopReceiving ()`

Privilege: Public

No Inheritance

Used to stop notifications from being sent to the callback. No notifications can be received in this state.

`void ~DsClNotificationReceiver ()`

Privilege: Public

No Inheritance

Used to destroy a notification receiver.

5.14.2.7 Class DsClQuery

Synopsis:

Parent Class: DsClCommand.

Is Not A Distributed Object

Is Associated With:

This class is derived from the class DsClCommand.

Description:

This public, local object simplifies the passing of query information from the client to the dataserver. The object is created in client space. The contents of the object will be used to create a Request object which will be passed to the dataserver. It is assumed that the "from" clause of an SQL query is inherent in specification of the dataserver to which the query is issued, that is, that the query is against "the inventory" of the dataserver. Any conversion to actual table names which may be necessary is done transparently to the client software.

Attributes:

Constraints

Privilege: Private

Data Type: GIParameterList

Default Value: NOT IDENTIFIED

No Inheritance

This attribute represents the constraints of a query (i.e. the "where" clause of an SQL statement).

MaxHitsB

Privilege: Private

Data Type: int

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the maximum number of granules to return (regardless of the number of actual hits, i.e. the granules which satisfy the search criteria). IN RELEASE B, this can be used to indicate the amount to check "estimated hits" against.

QueryTypeB

Privilege: Private

Data Type: DsTQueryType

Default Value: Inventory

No Inheritance

This attribute controls the scope of the query. DsTQueryType is an enum, where the possible query types are Inventory and WorkingCollection. Issuing a query against the inventory results in a search of the Inventory object. Issuing a query against the WorkingCollection results in a refinement of the existing collection. Issuing a query against an empty WorkingCollection will return null.

myLocalCallback

Privilege: Private

Data Type: GICallback

Default Value: NOT IDENTIFIED

No Inheritance

This attribute identifies the local callback routine to be invoked when any status is returned relative to this query. This attribute is passed to the DsCIRequest object which is created from the DsCIQuery object. The DsCIRequest object will use this information to relate status from the DsCISubmittedRequest to the DsCIRequest object. As far as the DsCIQuery object is concerned, this is just a pass-through.

myStatus

Privilege: Private

Data Type: GIStatus

Default Value: NOT IDENTIFIED

No Inheritance

This attribute allows the object to maintain information on current status.

Operations:

`DsClRequest & ConvertToCommand (DsClRequest &)`

Privilege: Protected Operation

No Inheritance

This operation converts the information in the Query object to a command format that is interpretable by the dataserver. The command is then packaged in a Request object prior to returning.

`void DsClQuery (GlParameterList &)`

Privilege: Public

No Inheritance

This version of the constructor takes the Constraints in the form of a ParameterList and uses the default for the AttributesToReturn.

`void DsClQuery ()`

Privilege: Public

No Inheritance

The default constructor creates an empty object.

`GlStatus & GetCallback (GlCallback &)`

Privilege: Public

No Inheritance

This operation allows the client software to get the current setting of the query callback.

`GlStatus & GetConstraints (GlParameterList &)`

Privilege: Public

No Inheritance

This operation returns the currently set list of user-defined query constraints.

`GlStatus & GetMaxHitsB (int &)`

Privilege: Public

No Inheritance

This operation returns the current setting of MaxHits. For Release A, MaxHits indicates the maximum number of rows to return (regardless of the actual number found by the query). This is in keeping with the definition of MaxHits in V0. In RELEASE B, it may be used to indicate the cutoff limit for running a query, i.e. if the estimated number of hits is greater than MaxHits, then don't run the query.

DsTQueryType GetQueryTypeB ()

Privilege: Public

No Inheritance

This operation returns the current setting for the QueryType.

G1Status & SetCallback (G1Callback &)

Privilege: Public

No Inheritance

This operation allows the client software to specify a local callback for queries so that the DsCIESDTRReferenceCollector object (which is distributed) can return interim status information to the client software.

G1Status & SetConstraints (G1ParameterList &)

Privilege: Public

No Inheritance

This operation allows the client software to pass a list of user- specified query constraints.

G1Status & SetMaxHitsB (int)

Privilege: Public

No Inheritance

This operations allows the client software to specify the number of rows to return in response to a query (this per the current definition of MaxHits in V0). IN RELEASE B, it can be used to specify the number of hits to compare to a query estimator; in that case the query will not execute if the anticipated number of result objects is greater than the value in MaxHits.

G1Status & SetQueryTypeB (DsTQueryType)

Privilege: Public

No Inheritance

This operation sets the scope of the query, i.e., run against the Inventory or against the WorkingCollection.

void ~DsClQuery ()

Privilege: Public

No Inheritance

The DsCLQuery's destructor.

5.14.2.8 Class DsClRequest

Synopsis:

Parent Class: DsSrRequestBase
Is Not A Distributed Object
Is Associated With:
Class: DsClSubmittedRequest(Private)
DsClAction (Aggregation)
DsClRequestVector (Aggregation)

Description:

A specialization of DsRequest for client interfaces. Allows the client to compose a request and submit it to the data server. Once submitted, the status may be polled, or a callback can be provided that is triggered on every status change.

Attributes:

myInfo

Privilege: Protected Attribute
Data Type: DsSrRequestInfo*
Default Value: NOT IDENTIFIED
Inherited From: DsSrRequestBase
A pointer to the core request information for this request. (In OO terms, the implementation for this interface).

mySubmittedRequest

Privilege: Protected Attribute
Data Type: DsClSubmittedRequest *
Default Value: NULL
Inherited From: DsSrRequestBase
The DsClSubmittedRequest that is associated with this request. Value is NULL until one is created/found.

myCallback

Privilege: Private
Data Type: GICallback
Default Value: NOT IDENTIFIED
No Inheritance
Callback that will be made anytime the status changes.

myCollector

Privilege: Private
Data Type: DsClESDTRreferenceCollector *
Default Value: NOT IDENTIFIED
No Inheritance
Reference to the collector to which this request belongs.

myLastResultsB

Privilege: Private

Data Type: GIParameterList&

Default Value: NOT IDENTIFIED

No Inheritance

This attribute stores the last set of results from the server side. This is in support of suspending a session. This operation allows the DsCIRequest to respond to a GetResults request even after it has been disconnected (normally the results information is stored on and retrieved from the server-side).

myLastStatusB

Privilege: Private

Data Type: GIStatus

Default Value: NOT IDENTIFIED

No Inheritance

This attribute stores the last status received from the server-side with respect to this request. This is to support the suspension of a session. It allows a disconnected request to provide status information to a user (normally this information would be stored on and retrieved from the server side).

myQuery

Privilege: Private

Data Type: DsCIQuery *

Default Value: NOT IDENTIFIED

No Inheritance

This is the query object associated with this request. The query object is needed during a search request because the collector needs to look up the callback for the query after it is called-back by the request when the search status changes.

myStateB

Privilege: Private

Data Type: DsECIState

Default Value: Active

No Inheritance

This attribute is maintained to support the ability to suspend and resume sessions. When a session is suspended, this attribute is set to "suspended" so that the DsCIESDTRreferenceCollector knows not to accept any more requests.

mySubmittedFlag

Privilege: Private

Data Type: RWBoolean

Default Value: NOT IDENTIFIED

No Inheritance

Indicates whether this request has been submitted.

Operations:

void DsSrRequestBase (DsESrRequestPriority = NORMAL)

Privilege: Public

Inherited From: DsSrRequestBase

Constructs a request with the given priority (LOW, NORMAL, HIGH).

void EstimateB ()

Privilege:

Inherited From: DsSrRequestBase

This operation provides information about the anticipated resources consumed by the request.

GLURVector & GetDomain ()

Privilege: Public

Inherited From: DsSrRequestBase

Returns the current UR vector of this request.

DsSrRequestInfo* GetInfo ()

Privilege: Protected Operation

Inherited From: DsSrRequestBase

Returns a pointer to the underlying data (DsSrRequestInfo) for this request.

DsESrRequestPriority GetPriority ()

Privilege: Public

Inherited From: DsSrRequestBase

Returns the current priority of the request.

DsClSubmittedRequestID GetSRID ()

Privilege: Protected Operation

Inherited From: DsSrRequestBase

Used to obtain the ID of the submitted request associated with this request.

`const GlStatus & GetStatus ()`

Privilege: Public

Inherited From: DsSrRequestBase

Returns a reference to the current status of the request.

`void SetDomain (GlURVector &)`

Privilege: Public

Inherited From: DsSrRequestBase

Used to set the UR vector for this request.

`void SetPriority (DsESrRequestPriority)`

Privilege: Public

Inherited From: DsSrRequestBase

Used to set the priority of the request.

`void ~DsSrRequest ()`

Privilege: Public

Inherited From: DsSrRequestBase

Destroys a request.

`GlStatus CancelB ()`

Privilege: Public

No Inheritance

Used to cancel a request. If the request is not executing (i.e. it's queued), then it is removed from the queue. If it is executing, it will be terminated after the current command completes.

`void DsClRequest ()`

Privilege: Public

No Inheritance

Used to construct an "empty" request, which should be filled-in by calling SetPriority (and other functions) and adding commands.

`void DsClRequest (cmd: DsClCommand *, pty:
DsESrRequestPriority)`

Privilege: Public

No Inheritance

Constructs a request with the given command and priority.

`const DsSrCost EstimateB ()`

Privilege: Public

No Inheritance

This operation estimates the cost, in terms of resources needed, to execute this request.

`DsClQuery * GetQuery ()`

Privilege: Public

No Inheritance

Returns the current query object associated with this request.

`const GlParameterList & GetResults ()`

Privilege: Public

No Inheritance

Returns the current results list from the associated DsClSubmittedRequest.

`DsEClState GetStateB ()`

Privilege: Public

No Inheritance

This operation returns the value of myState.

`const GlStatus& SetLastStatusB (GlStatus)`

Privilege: Public

No Inheritance

This operation is used to set the value of myLastStatus.

`void SetQuery (DsClQuery *)`

Privilege: Public

No Inheritance

Used to set the query object associated with this request. The query object is needed during a search request because the collector needs to look up the callback for the query after it is called-back by the request when the search status changes.

`void SetStateB (DsEClState)`

Privilege: Public

No Inheritance

This operation assigns a value to the myState attribute.

`void SetStatusCallback (GlCallback *)`

Privilege: Public

No Inheritance

Used to provide a client entry point to be called on every change of the status of the request.

`GlStatus Submit (DsClESDTReference &)`

Privilege: Public

No Inheritance

Used to submit a request to be executed by a single, specific ESDT. The request is in turn submitted to the "implied" DsClESDTReferenceCollector, i.e. the one the DsClESDTReference holds a pointer to.

`RWCString Textify ()`

Privilege: Public

No Inheritance

Used to convert a request into a human-readable format.

`void ~DsClRequest ()`

Privilege: Public

No Inheritance

Used to destroy a request.

`const GlStatus& DisconnectB ()`

Privilege: Public

No Inheritance

This operation puts the DsClRequest into suspended state and does the clean-up of the DsClSubmitted request (storing the previously server-side information in the local, client DsClRequest).

`GlParameterList& GetLastResultsB ()`

Privilege: Public

No Inheritance

This operation allows clients to get the final results from disconnected requests.


```
const GlStatus& GetLastStatusB ( )
```

Privilege: Public

No Inheritance

This operation is used to access the current value of the attribute myLastStatus. This is to support the suspension of sessions, i.e., a DsClRequest will be able to honor calls to GetStatus even after it has been disconnected (normally, the status information would be stored on and retrieved from the server-side).

```
const GlStatus& SetLastResultsB (GlParameterList&)
```

Privilege: Public

No Inheritance

This operation allows the DsClRequest to store the final results from a disconnected request in the DsClRequest object instead of the DsClSubmittedRequest object. This operation does a copy of the results from the server-side distributed object to the local DsCLRequest object.

```
GlStatus Submit (DsClESDTReferenceCollector &, GlURVector * =  
NULL)
```

Privilege: Public

No Inheritance

Used to submit a request that is to be executed over an entire collector, e.g. a search that adds/removes ESDTs or operation(s) to be performed by all of the ESDTs. A request can be submitted to a specific subset of ESDTs in the collection by passing a GlURVector with the URs of the desired ESDTs.

5.14.2.9 Class DsClRequestVector

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

DsClESDTReferenceCollector (Aggregation)

Description:

An instantiation of RWVector over DsClRequest. The operations and attributes of this class are defined by the definition of RWVector.

Attributes:

Operations:

5.14.2.10 Class DsClSubscription

Synopsis:

Parent Class: DsShSubscription
Is Not A Distributed Object
Is Associated With:
Class: DsClAction(Public) actionof
DsClSubscriptionCollector (Aggregation)

Description:

This class is the client side subscription which can either be created from advertisements or from existing subscriptions from the server side (through a stream.)

Attributes:

myAction

Privilege: Protected Attribute
Data Type: DsClAction
Default Value: NOT IDENTIFIED
No Inheritance
The action to be performed when the subscription fires.

myCollector

Privilege: Protected Attribute
Data Type: DsClSubscriptionCollector&
Default Value: NOT IDENTIFIED
No Inheritance
A pointer to the collector that this reference is a member of. If this pointer is null, then this reference is a member of one of the collectors in the static collector vector.

myDescription

Privilege: Protected Attribute
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
String which contains service of the subscription.

myDurationType

Privilege: Protected Attribute
Data Type: enum DsEClSubscriptionType
Default Value: {ONCE, OUTSTANDING}
No Inheritance
Time duration of subscriptions (i.e., can be done one time or forever (outstanding)).

myExpirationDate

Privilege: Protected Attribute

Data Type: RWDate

Default Value: NOT IDENTIFIED

No Inheritance

Identifies when this subscription will expire and be removed from the system. The value may be "never" (i.e. the subscription is permanent)

mySubmittedFlag

Privilege: Protected Attribute

Data Type: RWBoolean

Default Value: RWTrue

No Inheritance

Flag which shows whether the subscription has been submitted or not.

myUserInfo

Privilege: Protected Attribute

Data Type: DsClClient&

Default Value: NOT IDENTIFIED

No Inheritance

Client information, provided by client software.

ourCollectorvector

Privilege: Protected Attribute

Data Type: DsClSubscriptionCollectorVector&

Default Value: NOT IDENTIFIED

No Inheritance

Static vector of pointers to DsClSubscriptionCollector objects, one per dataserver.

Operations:

```
void DsClSubscription (userinfo, Advertisement&,
DsClSubscriptionCollector&)
```

Privilege: Public

No Inheritance

Constructor for client software (therefore, public) which gets attribute information from advertisements, such as service provider. If no collector has been provided, it goes and finds one, based on the static nature of the collector.

```
void DsClSubscription (submittedflag,  
DsClSubscriptionCollector&, Stream)
```

Privilege: Public

No Inheritance

Constructor for already existing collector which gets already existing subscriptions from the sever side through a stream.

```
void DsClSubscription ()
```

Privilege: Public

No Inheritance

The default constructor creates an empty object.

```
void GetAction (DsClAction &)
```

Privilege: Public

No Inheritance

Means of accessing myAction attribute (object), which will be communicated to the server what this subscription should do when it fires.

```
RWCString GetDescription ()
```

Privilege: Public

No Inheritance

Returns description, containing the service, as a RogueWave string.

```
DsEClSubscriptionType GetDurationtype ()
```

Privilege: Public

No Inheritance

Accesses myDurationType attribute as to whether subscriptions are done one time or forever (outstanding).

```
RWDate GetExpirationdate ()
```

Privilege: Public

No Inheritance

Public access to myExpirationDate attribute, which provides the expiration date of the subscription.

```
RWBoolean GetSubmittedflag ()
```

Privilege: Public

No Inheritance

Public access to flag as to whether a subscription has been submitted.

```
void GetUserinfo (GLClient&)
    Privilege: Public
    No Inheritance
    Public access to user information which can be put into the DsSrClient
    object.
```

```
void SetAction (DsClAction&)
    Privilege: Public
    No Inheritance
    Sets the myAction attribute for this particular subscription as
    determined by the client software.
```

```
void SetDescription (RWCString)
    Privilege: Public
    No Inheritance
    Allows the client software to fill in the Description attribute with
    service information.
```

```
void SetDurationType (DsEClSubscriptionType)
    Privilege: Public
    No Inheritance
    Sets the attribute which determines the existence type of the
    subscription.
```

```
void SetExpirationDate (RWDate)
    Privilege: Public
    No Inheritance
    Sets the expiration date of the subscription itself.
```

```
void SetSubmittedFlag (RWBoolean)
    Privilege: Protected Operation
    No Inheritance
    Sets the flag which indicates whether or not the DsClSubscription has
    actually been submitted to the dataserver (i.e. the client software is
    finished with filling in the information, and has invoked the Submit
    method).
```

```
GLStatus& Submit ()
    Privilege: Public
    No Inheritance
    Submits subscription to the subscription collector.
```

GlStatus& Withdraw ()
Privilege: Public
No Inheritance
Deletes a subscription from the subscription collector.

void ~DsClSubscription ()
Privilege: Public
No Inheritance
The DsClSubscription's destructor.

5.14.2.11 Class DsClSubscriptionCollector

Synopsis:

Parent Class: DsClCollector
Parent Class: DsClGenConnector
Distributed Object
Is Associated With:
This class is derived from the class DsClGenConnector

Description:

This public, distributed class is a specialization of the Collector class which handles DsClSubscriptions. This class provides, in addition to the normal vector operations, the ability to create a list of all subscriptions for a given user or advertisement, and a means of submitting and cancelling subscriptions. There are no attributes for this object.

Attributes:

myStatus
Privilege: Private
Data Type: GlStatus
Default Value: NOT IDENTIFIED
No Inheritance
This attribute allows the object to maintain information on current status.

Operations:

const GlStatus & BuildList (Advertisement&)
Privilege: Public
No Inheritance
This operation creates a list of all subscriptions for a given event.

const GlStatus & BuildList (MSS_UserProfile &)
Privilege: Public
No Inheritance
This operation creates a list of all subscriptions for a given user.

```
const GlStatus & BuildListB ()
```

Privilege: Public

No Inheritance

This operation allows ops/admin staff to get a list of all subscriptions in the system.

```
const GlStatus & CancelSubscription (DsClSubscription*)
```

Privilege: Private

No Inheritance

This operation creates a request to cancel the specified subscription.

```
DsClSubscription* CreateSubscription (RWBoolean  
SubmittedFlag, istream &Stream, DsClSubscriptionCollector  
*me)
```

Privilege: Private

No Inheritance

This is a private service used to build the set of subscriptions contained by the SubscriptionCollector.

```
void DsClSubscriptionCollector (GlUR &dataserver,  
MSS_UserProfile &)
```

Privilege: Public

No Inheritance

The constructor for DsClSubscriptionCollector's. This constructor establishes a set of Subscriptions for the user based on the provided science data server and the user information.

```
void ~DsClSubscriptionCollector ()
```

Privilege: Public

No Inheritance

The DsClSubscriptionCollector's destructor.

```
void DsClCollector (GlUR &dataserver, MSS_UserProfile &)
```

Privilege: Private

Inherited From: DsClCollector

The default constructor for this class is private so that an object of this class can not be created without providing enough information to establish a connection (i.e., either a dataserver ID or a session ID).

DsESrConnectionID GetConnectionID ()

Privilege: Public

Inherited From: DsClCollector

This public operation allows client software to retrieve the dataserver-assigned session ID so that users can turn off their terminals during long-running request processing, then log on later and reconnect to the session.

const G1Status & SubmitToServer (DsClRequest&)

Privilege: Protected Operation

Inherited From: DsClCollector

This protected operation passes a request from a DsClRequest to the dataserver after the DsClRequest has established the DsClSubmittedRequest to handle the distributed callback. This operation is fully implemented on the server side (i.e. in either DsSrConnection or the specialized connection object DsSrSession). All return values are handled by the DsClRequest object, which contains imbedded return parameters for this purpose.

void ~DsClCollector ()

Privilege: Public

Inherited From: DsClCollector

5.14.2.12 Class DsGeTypeID

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

DsClDescriptor (Aggregation)

DsClTypeInfo (Aggregation)

DsDeESDtdDescriptor (Aggregation)

Description:

This object uniquely identifies each ESDT's type. The type consists of a type name and a version number. Each type-version number pair is assigned a unique code. The set of all TypeIDs is stored persistently in a database.

Attributes:

myCode

Privilege: Private

Data Type: unsigned long

Default Value: NOT IDENTIFIED

No Inheritance

A unique number assigned to this type/version combination. Presumably it will be more efficient to use this number instead of the name and version to identify the type of an ESDT.

myName

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The name of this type. This name along with the version number is enough to uniquely identify this type. An example of a type name is CER03.

myVersionB

Privilege: Private

Data Type: RWCString

Default Value: NOT IDENTIFIED

No Inheritance

The version number of this ESDT. Version numbers are needed because an ESDT of the same name may have several implementations.

Operations:

```
void DsGetTypeID (RWCString &theCode)
```

Privilege: Public

No Inheritance

This constructor creates a TypeID instance from the given code. This code uniquely identifies a TypeID.

```
void DsGetTypeID (RWCString &theName, RWCString  
theVersion=NULL)
```

Privilege: Public

No Inheritance

This constructor takes the name of the type and an optional version number which it uses to create a TypeID. If the version number is not given, then the TypeID is created using the highest version for the given name.

`unsigned long & GetTypeCode ()`

Privilege: Public

No Inheritance

Returns the code for this TypeID.

`RWCString GetTypeNames ()`

Privilege: Public

No Inheritance

Returns the myName attribute which is the type name of this instance.

`RWCString GetTypeVersionB ()`

Privilege: Public

No Inheritance

Returns the string contained in the myVersion attribute. This is the version of this TypeID.

`void SetTypeCode (unsigned long)`

Privilege: Private

No Inheritance

This is a private member function used to set the value of the myCode attribute.

`void SetTypeNames (RWCString &)`

Privilege: Private

No Inheritance

This is a private member function used to set the value of the myName attribute.

`void SetTypeVersionB (RWCString &)`

Privilege: Private

No Inheritance

This is a private member function used to set the value of the myVersion attribute.

`void ~DsGetTypeID ()`

Privilege: Public

No Inheritance

The destructor for this type has no implementation at this time.

5.14.2.13 typedef DsTCIRequestCallback

GlCallback should be translated to refer to the typedef, DsTCIRequestCallback. Its basic capability is to represent a function pointer to an entry point in the client application. Control will be passed to this function point upon the asynchronous activity for which the “callback” was identified.

5.15 Storage Resource Management Classes

5.15.1 Storage Resource Management Classes Overview

The data server must be able to respond to policy changes regarding archiving of data vs. creation of it when needed, or priorities in the allocation of resources, such as working storage, staging disks, and the variety of persistent storage in the HSM. This requires that the data server be "policy neutral". The facilities to adapt the physical storage of data in the data server to policy, while minimizing impact to availability, is provided by the Storage Management CSCI (STMGT CSCI). This CSCI provides an isolation layer between the search and access views of the archived data in the clients domain, and the physical storage mechanisms of the data internal to the archive. Through the use of unique data identifiers, the STMGT CSCI externalizes its data holdings to the SDSRV CSCI, while hiding the actual physical storage of its data. This allows the STMGT CSCI to optimize its archive storage and data migration strategies, while maintaining a consistent reference to the data for its clients.

5.15.2 Storage Resource Management Class Descriptions

5.15.2.1 Class DsStArchive

Synopsis:

Parent Class: DsStStorageResource
Distributed Object
Is Associated With:
This class is derived from the class DsStStorageResource

Description:

To provide storage for persistent data. The Archive is the repository for all the permanently stored data in the DataServer.

Attributes:

myBackupB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
The identity of the backup archive.

myDataTypeNames

Privilege: Public
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
This attribute identifies the data type serviced by this archive.

myID

Privilege: Public
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
This is the Identity of the (primary) archive.

myNumDataTypes

Privilege: Public
Data Type: EcTShortInt
Default Value: 1
No Inheritance
This attribute indicates the number of data types serviced by this archive.

myOffsiteLocationB

Privilege: Private
Data Type: RWCString
Default Value: NOT IDENTIFIED
No Inheritance
The name of the offsite location for storage of offsite backups.

myResourceName

Privilege: Private
Data Type: RWCString *
Default Value: NOT IDENTIFIED
Inherited From: DsStResource
This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private
Data Type: *DsStResourceManager
Default Value: NOT IDENTIFIED
Inherited From: DsStResource
This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private
Data Type: RWCString *
Default Value: NOT IDENTIFIED
Inherited From: DsStResource
This attribute indicates the name of the resource pool schedule file.

Operations:

`EcUtStatus CopyFileB (SrcFilename: RWCString, DestFilename: RWCString)`

Privilege: Public

No Inheritance

This operation copies a file from one archive to a different archive.

`EcTLongInt CreateCheckSum (filename: RWCString)`

Privilege: Public

No Inheritance

This service creates a file checksum.

`EcUtStatus DeleteArchiveFileB (filename: RWCString)`

Privilege: Public

No Inheritance

This private service deletes a file from the archive serviced by the Resource Manager.

`GISTatus DeleteFileB (Filename: RWCString)`

Privilege: Public

No Inheritance

This public service provides clients of Storage Management the capability to delete a data file from the archive.

`void DsStArchiveB (primaryarchive: RWCString, backuparchive: RWCString, offsitelocation: RWCString)`

Privilege: Public

No Inheritance

This is the constructor for archive object.

`void EstimateDelayB (filename: RWCString)`

Privilege: Public

No Inheritance

This service provides an estimate of the time delay associated with accessing the archive. It provides and reports this time in seconds and reflects anticipated service times.

`EcTLongInt GetCheckSum (GIPParameterList*)`

Privilege: Public

No Inheritance

This service gets a file's previously calculated checksum.

void GetDataTypeName (ndtype: EcTShortInt, dtypename: RWCString *)

Privilege: Public

No Inheritance

This operation gets the name of the data types serviced by this archive.

GlStatus GetID (archiveID: RWCString)

Privilege: Public

No Inheritance

This operation gets the name of this archive.

GlStatus GetNumDataTypes (numdatatypes: EcTShortInt)

Privilege: Private

No Inheritance

This operation gets the number of data types serviced by the archive.

EcTVoid ReportChecksumErrorB ()

Privilege: Private

No Inheritance

This service reports gathered statistics on check sum errors encountered while retrieving data files from the archive.

ostream&outfile ReportPerformanceB ()

Privilege: Private

No Inheritance

This service reports performance statistics gathered for the requests accessing the archive.

EcUtStatus RestoreFileB (Filename: RWCString)

Privilege: Public

No Inheritance

This operation restores a file in a designated archive.

EcUtStatus Retrieve (RetrieveParams: DsStFileParametersB, Cost, *DsUzCostB)

Privilege: Public

No Inheritance

This operation retrieves specified files from the archive.

GlStatus SetDataTypeName (ndtype: EcTShortInt, dtypename: RWCString)

Privilege: Private

No Inheritance

This operation sets the name(s) of the data types serviced by this archive.

GlStatus SetID (archiveID: RWCString)

Privilege: Private

No Inheritance

This operation sets the name of the archive.

GlStatus SetNumDataTypes (numdatatypes: EcTShortInt)

Privilege: Private

No Inheritance

This operation sets the number of data types serviced by this archive.

EcUtStatus Store (StoreParams: DsStFileParametersB, Cost, *DsUzCostB)

Privilege: Public

No Inheritance

This operation stores data files into the archive.

EcTVoid ~DsStArchive (primaryarchive: RWCString, backuparchive: RWCString, offsitelocation: RWCString)

Privilege: Public

No Inheritance

This is the destructor for the archive object.

secs:EcTFloat EstimateDelayB (Filename: RWCString)

Privilege: Public

Inherited From: DsStStorageResource

This service provides an estimate of the time delay associated with retrieving a file from the archive.

EcTPtr Allocate (ResourceTypr: RWCString *, MediaType: EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size: EcTLongInt, ProfileInfo: ptr)

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum, DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

`EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)`

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

`G1Status RestoreResourceInformation ()`

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EcTVoid SetMediaSize (MediaSize: EcTEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`G1Status SetPriority (RequestID: EcTInt, NewPriority:
EcTShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`G1Status SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`G1Status SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EcTVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.15.2.2 Class DsStCDROM

Synopsis:

Parent Class: DsStPhysicalResource

Distributed Object

Is Associated With:

This class is derived from the class DsStPhysicalResource

Description:

This class provides an interface to the CD-ROM resource. The Ingest Client to Storage Management can use CDROM devices for ingesting data. The Data Distribution Client to Storage Management can use these same resources for distributing data to requesting users.

Attributes:

myCapacity

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the capacity (in KBYTES) of the media currently mounted.

myFirstUsedSector

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the first used sector on the volume currently mounted in the CDROM.

myNextFreeDataSector

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the next free data sector on the volume currently mounted in the CDROM.

myNextFreeDirSector

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the next free directory sector on the volume currently mounted in the CDROM.

myRemainingSectors

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the number of unused (i.e., unwritten) sectors remaining on the media.

myResourceName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

Operations:

GlStatus Dismount (VolName: RWCString *)

Privilege: Public

No Inheritance

This public operation dismounts the specified volume from a CDROM resource.

GlStatus GetCapacity (MaxBytes: EcTLongInt)

Privilege: Public

No Inheritance

This operation gets the capacity of the volume currently mounted in the CDROM resource.

GlStatus GetFirstUsedSector (FirstSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation gets the value of the first used sector on the volume currently mounted in the CDROM.

GlStatus GetNextFreeDataSector (NextSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation gets the next free data sector on the volume currently mounted in the CDROM.

GlStatus GetNextFreeDirSector (NextSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation gets the next free directory sector on the volume currently mounted in the CDROM resource.

GlStatus GetRemainingSectors (AvailSectors: EcTLongInt)

Privilege: Public

No Inheritance

This public service gets the number of free unwritten sectors on the media currently mounted in the allocated CDROM device.

GlStatus Mount (VolName: RWCString *)

Privilege: Public

No Inheritance

This public operation mounts the specified volume on the CDROM resource.

GlStatus SetCapacity (MaxBytes: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the capacity (in KBYTES) of the media currently mounted in the CDROM resource.

GlStatus SetFirstUsedSector (FirstSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the first used sector used on the volume currently mounted in the CDROM.

GlStatus SetNextFreeDataSector (NextSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the value of the next free data sector of the volume currently mounted in the CDROM.

GlStatus SetNextFreeDirSector (NextSector: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the value of the next free sector on the volume currently mounted in the CDROM resource.

GlStatus SetRemainingSectors (AvailSectors: EcTLongInt)

Privilege: Public

No Inheritance

This operation sets the number of free and unwritten sectors remaining on the media currently mounted on the CDROM device.

void ~DsStCDROM ()

Privilege: Public

No Inheritance

This is the destructor for the CDROM object.

EcTPtr Allocate (ResourceTypr: RWCString *, MediaType:
EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size:
EcTLongInt, ProfileInfo: ptr)

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation:
RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum,
DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

GlStatus RestoreResourceInformation ()

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EctVoid SetMediaSize (MediaSize: EctEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`GlStatus SetPriority (RequestID: EctInt, NewPriority: EctShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`GlStatus SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`GlStatus SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EctVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.15.2.3 Class DsStFaxB

Synopsis:

Parent Class: DsStPhysicalResource

Distributed Object

Is Associated With:

This class is derived from the class DsStPhysicalResource

Description:

This class provides an interface to FAX resources managed by the Resource Manager.

Attributes:

myResourceName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

Operations:

void ~DsStFaxB ()

Privilege: Public

No Inheritance

This is the destructor for the FAX interface object.

EcTPtr Allocate (ResourceTypr: RWCString *, MediaType:
EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size:
EcTLongInt, ProfileInfo: ptr)

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum, DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *, SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

`GlStatus RestoreResourceInformation ()`

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EctVoid SetMediaSize (MediaSize: EctEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`GlStatus SetPriority (RequestID: EctInt, NewPriority: EctShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`GlStatus SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`GlStatus SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EctVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.15.2.4 Class DsStNetworkResource

Synopsis:

Parent Class: DsStResource

Distributed Object

Is Associated With:

This class is derived from the class DsStResource

Description:

This class provides a push/pull interface to the network for transferring data for both ingest and data distribution operations.

Attributes:

`myDestination`

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute identifies the destination machine for the network data transfer.

`mySource`

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute identifies the source machine for the network data transfer.

`myResourceName`

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

`ourManager`

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

`ourSchedule`

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

Operations:

`RWCString * GetDestination ()`

Privilege: Public

No Inheritance

This operation gets the name of the destination machine for the network data transfer.

`RWCString * GetSource ()`

Privilege: Public

No Inheritance

This operation gets the name of the source machine for the network data transfer.

`EcTVoid SetDestination (NodeName: RWCString *)`

Privilege: Public

No Inheritance

This operation sets the name of the destination machine for the network data transfer.

`EcTVoid SetSource (NodeName: RWCString *)`

Privilege: Public

No Inheritance

This operation sets the name of the source machine for the network data transfer.

`EcTVoid ~DsStNetworkResource ()`

Privilege: Public

No Inheritance

This is the destructor for the `DsStNetworkResource` class.

`EcTPtr Allocate (ResourceTypr: RWCString *, MediaType: EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size: EcTLongInt, ProfileInfo: ptr)`

Privilege: Public

Inherited From: `DsStResource`

This public service allocates a specific device from a pool of resources.

`EcUtStatus Deallocate (ResourceRef: ptr)`

Privilege: Public

Inherited From: `DsStResource`

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum, DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *, SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

`GtStatus RestoreResourceInformation ()`

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EctVoid SetMediaSize (MediaSize: EctEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`GtStatus SetPriority (RequestID: EctInt, NewPriority: EctShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`GtStatus SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`GtStatus SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EctVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.15.2.5 Class DsStPrinter

Synopsis:

Parent Class: DsStPhysicalResource

Distributed Object

Is Associated With:

This class is derived from the class DsStPhysicalResource

Description:

This class provides an interface to the resource pool of printers used by data distribution to produce shipping labels, packing slips and media labels.

Attributes:

myResourceName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

Operations:

EcUtStatus Print (FileName: RWCString *)

Privilege: Public

No Inheritance

This operation writes text to the printer.

EcTVoid ~DsStPrinter ()

Privilege: Public

No Inheritance

This operation destroys the Printer object.

EcTPtr Allocate (ResourceTypr: RWCString *, MediaType:
EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size:
EcTLongInt, ProfileInfo: ptr)

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum, DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

`EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)`

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

`G1Status RestoreResourceInformation ()`

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EcTVoid SetMediaSize (MediaSize: EcTEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`G1Status SetPriority (RequestID: EcTInt, NewPriority:
EcTShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`G1Status SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`G1Status SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EcTVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.15.2.6 Class DsStRequestManager

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
Class: DsStResourceManager(Private) routes requeststo

Description:

This class services selected service requests from operations staff and clients. It insures that the service requests are properly routed to the appropriate resource manager(s).

Attributes:

myNumberRequests

Privilege: Private
Data Type: EcTUShortInt
Default Value: 0
No Inheritance

This attribute indicates the number of requests for service which have been received but are still in progress.

myRequestTable

Privilege: Private
Data Type: ptr
Default Value: NOT IDENTIFIED
No Inheritance

This attribute identifies the table of outstanding requests.

Operations:

EcUtStatus Abort (RequestID: EcTULongInt)

Privilege: Public
No Inheritance

This public service causes the termination of processing of the specified request if the request is currently queued.

EcUtStatus ChangeDeviceStatus (DeviceName: RWCString *,
DeviceStatus: RWCString *)

Privilege: Public
No Inheritance

This public service provides operations staff the capability to change the operational status of a specific resource device under control of the Storage Management CI.

EcUtStatus ChangeRequestPriority (RequestID: EcTULongInt,
Priority: EcTShortInt)

Privilege: Public

No Inheritance

This public service changes the priority of a queued request for service.
It has no effect on the request if the request is not currently queued.

EcTVoid DismountVolume (VolName: RWCString *)

Privilege: Public

No Inheritance

This operation dismounts a specific volume.

EcTVoid EndService (RequestID: EcTULongInt)

Privilege: Private

No Inheritance

This operation marks the end of service of a request. It removes the
entry in the request table which corresponds to the specified request id.

DsUzCostB EstimateCostsB (RequestSize: EcTInt, Service:
RWCString)

Privilege: Public

No Inheritance

This service provides an estimate of the cost in resources for requests
received from Storage Management clients.

TimeDelay: EcTFloat EstimateStorageAccessDelayB (Filename:
RWCString *)

Privilege: Public

No Inheritance

This service provides an estimate of the time delay to retrieve a file from
the archive. The reported units are in seconds.

EcUtStatus GetDeviceStatus (DeviceName: RWCString
*, DeviceStatus: RWCString *)

Privilege: Public

No Inheritance

This public service gets the operational status of a specific resource
device.

*DsStResourceManager LocateManager (RequestID: EcTULongInt)

Privilege: Private

No Inheritance

This operation identifies and provides the appropriate resource manager
for requests which are still in-progress.

EcUtStatus MountVolume (VolName: RWCString *)

Privilege: Public

No Inheritance

This operation mounts a specific volume.

ostream&outfile ReportConfigB (ConfigFile: EcTPtr)

Privilege: Public

No Inheritance

This public service displays information contained in a specified configuration file.

ostream&outfile ReportOperationsB (ResourceType: RWCString)

Privilege: Public

No Inheritance

This service provides a report on queuing and/or scheduling operations performed by the Resource Manager.

ostream&outfile ReportResourcePerformanceB ()

Privilege: Public

No Inheritance

This service provides a performance report for specified resources managed by the Resource Manager.

ostream&outfile ReportResourceStats (DeviceName: RWCString *)

Privilege: Public

No Inheritance

This public service provides various resource related reports for all or specific devices managed by the Resource Manager.

GlStatus RequestStatus (RequestID: EcTULongInt)

Privilege: Public

No Inheritance

This public service requests the status of a previously submitted data store or data retrieve request.

EcTVoid StartService (RequestID: EcTULongInt)

Privilege: Private

No Inheritance

This operation marks the start of service for a request. It places an entry in the Request Table which corresponds to the specified request id.

5.15.2.7 Class DsStReservation

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
Class: DsStResourceSchedule(Private) is reflected in

Description:

This class requests preallocation and/or reservation of a resource. The Reservation is an internal mechanism to allocate a resource at some future time. The reservation request identifies the requestor, the resource to be allocated and the future time the resource will be required.

Attributes:

myEndTime

Privilege: Public
Data Type: RWCString *
Default Value: NOT IDENTIFIED
No Inheritance
This attribute indicates the end time and date for the reservation.

myPriority

Privilege: Public
Data Type: EcTUShortInt
Default Value: 0
No Inheritance
This attribute indicates the priority of the reservation request.

myRequester

Privilege: Public
Data Type: RWCString *
Default Value: NOT IDENTIFIED
No Inheritance
This attribute identifies the reservation requester.

myReservationRef

Privilege: Public
Data Type: EcTULongInt
Default Value: 0
No Inheritance
This attribute identifies the reservation confirmation number for the resource.

myResourceType

Privilege: Public

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the kind of resource that is being reserved. Possible values include Staging Disk, Network Resource, 4mm Tape, 8mm Tape and CD-ROM.

mySize

Privilege: Public

Data Type: EcTUShortInt

Default Value: 1

No Inheritance

This attribute indicates the size (in the case of staging disk) or number of a specific resource desired to be reserved.

myStartTime

Privilege: Public

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the start time and date of the reservation.

mySubmitTime

Privilege: Public

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This is the time that the reservation request is submitted.

Operations:

GlStatus Activate (RequestID: EcTULongInt, ResourceRef: ptr)

Privilege: Public

No Inheritance

This public service activates the resource reservation (i.e., exchanges the resource reservation for a resource allocation).

GlStatus Cancel (RequestID: EcTULongInt)

Privilege: Public

No Inheritance

This public service cancels a previously made reservation for a resource.

G1Status Make ()

Privilege: Public

No Inheritance

This public service creates an initial blank reservation request.

G1Status SetEndTime (TimeDate: RWCString *)

Privilege: Public

No Inheritance

This public operation sets the desired end time and date for the resource reservation.

G1Status SetPriority (Priority: EcTUShortInt)

Privilege: Public

No Inheritance

This public service sets the priority of the reservation request.

G1Status SetRequester (RequesterName: RWCString)

Privilege: Public

No Inheritance

This operation sets the name of the requester of the resource reservation.

G1Status SetResSize (ReservationSize: EcTUShortInt)

Privilege: Public

No Inheritance

This public service sets the anticipated size of data (in KBYTES) to be transferred/handled by the resource being reserved.

G1Status SetResourceType (ResourceType: RWCString *)

Privilege: Public

No Inheritance

This public service sets the kind of resource that is being reserved.

G1Status SetStartTime (TimeDate: RWCString *)

Privilege: Public

No Inheritance

This operation sets the start time and date for the resource reservation.

G1Status Submit (RequestID: EcTULongInt)

Privilege: Public

No Inheritance

This public service submits the completed reservation request to reserve a resource for some time into the future.

5.15.2.8 Class DsStResource

Synopsis:

No Parent Class
Distributed Object
Is Associated With:
Class: DsStPullList(Private)
Class: DsStStagingDataList(Private)
Class: DsStResourceManager(Private) manages access to
Class: DsStResourceSchedule(Private)
provides scheduling information to

Description:

To provide an interface to the services provided by the resources of the data server. The Resource is the generic interface to the specific resources within the Data Server.

Attributes:

myResourceName
Privilege: Private
Data Type: RWCSString *
Default Value: NOT IDENTIFIED
No Inheritance
This attribute uniquely identifies the specific device in the resource pool.

ourManager
Privilege: Private
Data Type: *DsStResourceManager
Default Value: NOT IDENTIFIED
No Inheritance
This attribute indicates the Resource Manager for this resource pool.

ourSchedule
Privilege: Private
Data Type: RWCSString *
Default Value: NOT IDENTIFIED
No Inheritance
This attribute indicates the name of the resource pool schedule file.

Operations:

EctPtr Allocate (ResourceTypr: RWCSString *, MediaType:
EtCEnum, RequestID: EtTULongInt, Priority: EtTShortInt, Size:
EtTLongInt, ProfileInfo: ptr)
Privilege: Public
No Inheritance
This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

No Inheritance

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

No Inheritance

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum, DeviceTime: EcTUShortInt)

Privilege: Private

No Inheritance

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

No Inheritance

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

No Inheritance

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

No Inheritance

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

No Inheritance

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

No Inheritance

This public service transfers data files from specific ingest devices to an internal ECS storage device.

G1Status RestoreResourceInformation ()

Privilege: Private

No Inheritance

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

EcTVoid SetMediaSize (MediaSize: EcTEnum)

Privilege: Private

No Inheritance

This operation sets the size of the media for the current allocation of the resource device.

G1Status SetPriority (RequestID: EcTInt, NewPriority:
EcTShortInt)

Privilege: Private

No Inheritance

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

G1Status SetResourceID (ResourceName: RWCString *)

Privilege: Public

No Inheritance

This operation sets the name of the resource allocated.

G1Status SetStatus (DeviceStatus: RWCString *)

Privilege: Public

No Inheritance

This operation sets the status of a specific device of the resource.

EcTVoid ~DsStResource ()

Privilege: Public

No Inheritance

This is the destructor for the resource object.

5.15.2.9 Class DsStResourceConfig

Synopsis:

Parent Class: DsCnConfiguration
Distributed Object
Is Associated With:
This class is derived from the class DsCnConfiguration

Description:

This class is a specialization of the DsCnConfiguration base class. It manages configuration parameters for resources managed by the Storage Management CI.

Attributes:

Operations:

5.15.2.10 Class DsStSchedulingConfig

Synopsis:

Parent Class: DsCnConfiguration
Distributed Object
Is Associated With:
This class is derived from the class DsCnConfiguration

Description:

This class is a specialization of the DsCnConfiguration base class. It manages configuration parameters for scheduling resources managed by the Storage Management CI. Resources currently managed by the CI include 4mm Tape, 8mm Tape, 3480 tape, 3490 tape, CDROM, FAX, Staging Disk, the ECS Archive and the ECS communications network.

Attributes:

Operations:

5.15.2.11 Class DsStStagingDisk

Synopsis:

Parent Class: DsStStorageResource
Distributed Object
Is Associated With:
This class is derived from the class DsStStorageResource

Description:

To provide an interface to the staging disk resource for the temporary storage of data. The StagingDisk provides temporary and buffer storage.

Attributes:

myResourceName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

myAvailableSpace

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the amount of free disk space (in KBYTES) remaining for usage.

myMaxSize

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the maximum size (in KBYTES) of staging disk that can be allocated to the user at any one time.

myPathname

Privilege: Public

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This is the pathname to the root directory associated with this allocation of staging disk.

mySize

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the size (in KBYTES) of the staging disk allocated.

Operations:

secs:EcTFloat EstimateDelayB (Filename: RWCString)

Privilege: Public

Inherited From: DsStStorageResource

This service provides an estimate of the time delay associated with retrieving a file from the archive.

EcTPtr Allocate (ResourceTypr: RWCString *, MediaType: EtCEnum, RequestID: EcTULongInt, Priority: EcTShortInt, Size: EcTLongInt, ProfileInfo: ptr)

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

EcUtStatus Deallocate (ResourceRef: ptr)

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum,
DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

GlStatus RestoreResourceInformation ()

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

EctVoid SetMediaSize (MediaSize: EctEnum)

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

G1Status SetPriority (RequestID: EctInt, NewPriority: EctShortInt)

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

G1Status SetResourceID (ResourceName: RWCString *)

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

G1Status SetStatus (DeviceStatus: RWCString *)

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

EctVoid ~DsStResource ()

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

G1Status CopyFile (SourceFile: RWCString *, DestFile: RWCString *)

Privilege: Private

No Inheritance

This operation copies a file on staging disk.

G1Status ExtendStaging (ExtendSize: EctShortInt)

Privilege: Private

No Inheritance

This operation extends the size of an allocation of staging disk up to but not exceeding a hard limit. This hard limit is predetermined and set by operations personnel.

GlStatus GetAvailableSpace (AvailableSpace: EcTLongInt)

Privilege: Public

No Inheritance

This operation gets the amount of available disk space for an allocation.

GlStatus GetMaxSize (MaxSize: EcTLongInt)

Privilege: Public

No Inheritance

This operation gets the maximum size of the staging disk allocation.

GlStatus GetPathName (PathName: RWCString *)

Privilege: Public

No Inheritance

This operation gets the path name for the allocation of staging disk.

GlStatus GetSize (CurrentSize: EcTLongInt)

Privilege: Public

No Inheritance

This operation gets the current size of the staging disk allocation.

ostream&outfile ReportStatsB ()

Privilege: Public

No Inheritance

This service reports statistics gathered on that portion of staging disk which can be allocated as a work area (i.e., read/write staging disk).

GlStatus SetAvailableSpace (AvailableSpace: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the amount of available disk space for the allocation.

GlStatus SetMaxSize (MaxSize: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the maximum size for the allocation of staging disk.

GlStatus SetPathName (PathName: RWCString *)

Privilege: Public

No Inheritance

This operation sets the path name for the allocation of staging disk.

GlStatus SetSize (CurrentSize: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the current size of the staging disk allocation.

void ~DsStStagingDisk ()

Privilege: Public

No Inheritance

This operation is the destructor for the StagingDisk object.

5.15.2.12 Class DsStStream

Synopsis:

No Parent Class

Distributed Object

Is Associated With:

DsStStagingDisk (Aggregation)

Description:

This class provides access to the standard UNIX Fstreams.

Attributes:

myFileName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the name of the file created in the staging disk read/write area.

mySize

Privilege: Private

Data Type: EcTLongInt

Default Value: NOT IDENTIFIED

No Inheritance

This attribute indicates the size of the file in Kbytes.

Operations:

EcTVoid ~DsStStream ()

Privilege: Public

No Inheritance

This is the destructor for the DsStStream object.

5.15.2.13 Class DsStTape

Synopsis:

Parent Class: DsStPhysicalResource

Distributed Object

Is Associated With:

This class is derived from the class DsStPhysicalResource

Description:

This class provides an interface to tape resources managed by Storage Management. The Ingest Client to Storage Management can use tape resources to ingest data files. The Data Distribution Client to Storage Management can use tape resources to distribute data files to requesting users.

Attributes:

myBlockCount

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the current number of blocks which have been read from or written to the media currently mounted on the tape device.

myBlockSize

Privilege: Private

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the block size (in bytes) used for reading and writing to tape.

myCapacity

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the capacity (in KBYTES) of the media currently mounted in the tape resource.

myNextBlock

Privilege: Private

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the next block on the media which can be read or written to.

myRemainingBlocks

Privilege: Public

Data Type: EcTLongInt

Default Value: 0

No Inheritance

This attribute indicates the number of blocks remaining on the media.

myTapeType

Privilege: Public

Data Type: RWCString *

Default Value: NOT IDENTIFIED

No Inheritance

This attribute identifies the type of tape resource (e.g., 4mm, 8mm, 9track, 3480, 3490).

myResourceName

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute uniquely identifies the specific device in the resource pool.

ourManager

Privilege: Private

Data Type: *DsStResourceManager

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the Resource Manager for this resource pool.

ourSchedule

Privilege: Private

Data Type: RWCString *

Default Value: NOT IDENTIFIED

Inherited From: DsStResource

This attribute indicates the name of the resource pool schedule file.

Operations:

EcUtStatus Dismount (VolName: RWCString *)

Privilege: Public

No Inheritance

This public operation dismounts the specified volume from a tape device.

EcUtStatus GetBlockCount ()

Privilege: Public

No Inheritance

This operation gets the current number of blocks which have been read or written to the media currently mounted on the tape device.

EcUtStatus GetBlockSize ()

Privilege: Public

No Inheritance

This operation gets the block size (in bytes) used for read and write operations to the tape device.

EcTLongInt GetNextBlock ()

Privilege: Public

No Inheritance

This operation gets the next block to be read or written from the volume currently mounted on the tape resource.

EcTLongInt GetRemainingBlocks ()

Privilege: Public

No Inheritance

This public service gets the number of blocks remaining on the tape media which can be used to record data files.

EcUtStatus Mount (VolName: RWCString *)

Privilege: Public

No Inheritance

This public service mounts the specified volume onto a tape device.

EcUtStatus Rewind ()

Privilege: Public

No Inheritance

This public service rewinds the media currently mounted on the tape device.

EcTVoid SetBlockCount (BlockCount: EcTLongInt)

Privilege: Private

No Inheritance

This operation sets the number of blocks which have been currently read or written from or to the media currently mounted.

`EctVoid SetBlockSize (BlockSize: EctLongInt)`

Privilege: Private

No Inheritance

This operation sets the block size (in bytes) for read and write operations to the tape resource.

`EctVoid SetNextBlock (NextBlock: EctLongInt)`

Privilege: Private

No Inheritance

This operation sets the next block to be read or written from the volume currently mounted on the tape resource.

`EctVoid SetRemainingBlocks (AvailBlocks: EctLongInt)`

Privilege: Private

No Inheritance

This operation sets the number of blocks remaining which have been read from or written to the tape media.

`EctVoid ~DsStTape ()`

Privilege: Public

No Inheritance

This is the destructor for the DsStTape class.

`EctPtr Allocate (ResourceTypr: RWCString *, MediaType: EtCEnum, RequestID: EctULongInt, Priority: EctShortInt, Size: EctLongInt, ProfileInfo: ptr)`

Privilege: Public

Inherited From: DsStResource

This public service allocates a specific device from a pool of resources.

`EcUtStatus Deallocate (ResourceRef: ptr)`

Privilege: Public

Inherited From: DsStResource

This public service releases or frees a previously allocated resource.

`EcUtStatus DistFrom (SourceResource: RWCString *, Operation: RWCString *, DataItemName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This public service supports the transfer of data files from internal storage to specific distribution devices.

GlStatus GetDevTime (Size: EcTULongInt, MediaSize: EtCEnum,
DeviceTime: EcTUShortInt)

Privilege: Private

Inherited From: DsStResource

This operation gets a projected service time for an allocation of a resource. This service time is used in determining if the allocation request for a resource is queued or the resource is allocated.

EcTEnum GetMediaSize ()

Privilege: Public

Inherited From: DsStResource

This public service provides the size of the media for the allocated resource device.

EcTUShortInt GetPriority (RequestID: EcTULongInt)

Privilege: Public

Inherited From: DsStResource

This operation gets the current priority of the resource.

RWCString GetResourceID ()

Privilege: Private

Inherited From: DsStResource

This public service provides the name of the resource allocated.

RWCString GetStatus ()

Privilege: Private

Inherited From: DsStResource

This public service gets the status of a device in the resource pool.

EcUtStatus IngestFrom (SourceMachine: RWCString *,
SourceDirectory: RWCString *, DestMachine: RWCString)

Privilege: Public

Inherited From: DsStResource

This public service transfers data files from specific ingest devices to an internal ECS storage device.

GlStatus RestoreResourceInformation ()

Privilege: Private

Inherited From: DsStResource

This service will restore configurable parameters to preset values. This will be used in restart or cold start situations.

`EctVoid SetMediaSize (MediaSize: EctEnum)`

Privilege: Private

Inherited From: DsStResource

This operation sets the size of the media for the current allocation of the resource device.

`GtStatus SetPriority (RequestID: EctInt, NewPriority: EctShortInt)`

Privilege: Private

Inherited From: DsStResource

This operation sets the current priority of the resource to a specified value. The initial priority is set to the scheduling priority of the request.

`GtStatus SetResourceID (ResourceName: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the name of the resource allocated.

`GtStatus SetStatus (DeviceStatus: RWCString *)`

Privilege: Public

Inherited From: DsStResource

This operation sets the status of a specific device of the resource.

`EctVoid ~DsStResource ()`

Privilege: Public

Inherited From: DsStResource

This is the destructor for the resource object.

5.16 Universal Reference Classes

5.16.1 Universal Reference Classes Overview

Universal References (URs) provide applications and users a system wide mechanism for referencing ECS data and service objects. Once a UR is made for an object, the object can be disposed of and later reconstituted from the UR. URs can refer to objects that may be local to an address space, or remote.

URs are implemented as a framework which provides objects the capability to create URs for themselves, distribute URs throughout the system, then use these URs to reconstitute and/or access the original object. URs themselves are objects which can externalize themselves into an ASCII representation, then internalize from their ASCII representation back to objects. The content of the externalized UR is human transcribable and transport friendly. While the UR mechanism guarantees reliable data externalization and internalization, the content of each type of UR is application specific. Only the object that initially provides the UR (from now on we will call this the "UR Provider") is allowed to access and understand its content. URs are strongly typed to

enforce appropriate access control to internal data both at compile time and during runtime. Since URs are typed and have object specific data in them, separate UR object classes exist for each UR Provider object class referred to. All of these UR classes use the mechanisms provided by the UR framework. The framework also provides UR Providers support for their common requirements.

5.16.2 Universal Reference Class Descriptions

5.16.2.1 Class EcUrClassID

Synopsis:

No Parent Class
Is Not A Distributed Object
Is Associated With:
GrLiAnyURClass (Aggregation)

Description:

Attributes:

myRep

Privilege: Protection Not Identified
Data Type: NOT IDENTIFIED
No Inheritance
This attribute is the internal representation of the Class ID.

Operations:

EcUrClassID (void)

Privilege: Protection Not Identified
No Inheritance
This method is the default constructor for this class. This object's value, while initialized, is still undefined.

EcUrClassID (const int)

Privilege: Protection Not Identified
No Inheritance
This method will construct an object whose value is defined by the argument.

EcUrClassID (const char*)

Privilege: Protection Not Identified
No Inheritance
This method will construct an object whose value is defined by the argument.

`EcUrClassID (const EcUrClassID&)`

Privilege: Protection Not Identified

No Inheritance

This method is the copy constructor the object.

`IsValid (void)`

Privilege: Protection Not Identified

No Inheritance

`hash (void)`

Privilege: Protection Not Identified

No Inheritance

`operator!= (const EcUrClassID&)`

Privilege: Protection Not Identified

No Inheritance

This method is the inequality operator. It will return a non-zero value if the objects being compared are logically different.

`operator<< (ostream&)`

Privilege: Protection Not Identified

No Inheritance

`operator== (const EcUrClassID&)`

Privilege: Protection Not Identified

No Inheritance

`operator>> (istream&)`

Privilege: Protection Not Identified

No Inheritance

This is the stream extraction operator for the method. Unfortunately, while the name 'extraction' is similar to the UR operation 'Extract' its behavior is unrelated. This method reads the contents of the class ID from the input stream argument. It can only read a class ID written with the insertion operator<<. An exception will be raised if the stream or stream data is invalid.

`~EcUrClassID (void)`

Privilege: Protection Not Identified

No Inheritance

This method is the destructor for the object. State cleanup before the object is destroyed will occur here.

5.16.2.2 Class EcUrUR

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: EcUrURMaker(Public)

Class: EcUrURProvider(Public)

Description:

This is the abstract base class for all Universal Reference (UR)s. A UR is a special ECS identifier for an object. What makes it special is that an object can be identified, but the object does not have to exist in memory at the time. The contents of a UR are specified by subclasses. Generally speaking, the contents are the key elements of the object that this UR refers to. It can be thought of as DNA. We can reconstitute or clone an organism (i.e. object or URProvider) given its DNA (i.e. UR). The key public methods are "Externalize" and "Internalize"

Attributes:

Operations:

`EcUrUR (void)`

Privilege: Protection Not Identified

No Inheritance

This protected constructor initializes the object.

`Externalize (ostream&)`

Privilege: Protection Not Identified

No Inheritance

This method is the public interface for exporting the contents of the UR to an output stream. This data can be imported later with Internalize. This method allows URs to persist outside the runtime of the application. The stream will contain information (processed by Internalize) that prevents tampering.

`ExternalizeClassData (ostream&)`

Privilege: Protection Not Identified

No Inheritance

This method exports the UR's state to the output stream argument. This method shall be overridden by all concrete UR classes. The data written should have been read back from the stream with `InternalizeClassData`. This method shall call all appropriate associated objects' `ExternalizeClassData` method. An associated object is either a direct base class or a contained object. Note the public interface for this functionality is `Externalize`.

`GetURID (void)`

Privilege: Protection Not Identified

No Inheritance

This method returns the class ID of the UR object. Often abstract base class URs are passed around. This method can be used to find the actual concrete object being passed. The method `GetURProviderID` can be used to find out what concrete object this UR refers to.

`GetURProviderID (void)`

Privilege: Protection Not Identified

No Inheritance

The method returns the Class ID of the object that is referred to by this UR. This is the object that Extract'ed this UR and that can Reconstitute itself from this UR.

`Internalize (istream&)`

Privilege: Protection Not Identified

No Inheritance

This is the public method for importing data from a stream into a UR. If the stream does not contain the correct kind of data for this UR or if the data is invalid, an exception will be raised.

`InternalizeClassData (istream&)`

Privilege: Protection Not Identified

No Inheritance

This method imports the UR's state from the input stream argument. This method shall be overridden by all concrete UR classes. The data read should have been written to the stream with `ExternalizeClassData`. This method shall call all appropriate associated objects' `InternalizeClassData` method. An associated object is either a direct base class or a contained object. Note the public interface for this functionality is `Internalize`. Exceptions should be raised in this operation if errors occur.

`ReadTypingData (istream&, EcUrClassID&)`

Privilege: Protection Not Identified

No Inheritance

This private method allows a friend object class to determine the UR class ID contained in a stream. It is used by URMaker. The class ID is placed in the Class ID reference argument.

`operator<< (ostream&)`

Privilege: Protection Not Identified

No Inheritance

This method the same as the Externalize method except it support the standard stream insertion signature.

`operator>> (istream&)`

Privilege: Protection Not Identified

No Inheritance

This method the same as the Internalize method except it support the standard stream extraction signature.

`~EcUrUR (void)`

Privilege: Protection Not Identified

No Inheritance

This protected method is the destructor for the object. It will clean up state prior to the object being destroyed.

5.16.2.3 Class EcUrURMaker

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: EcUrUR(Public)

Description:

This class supports two correlated responsibilities. First, it is an object factory for Universal Reference (UR)s. It allows subclasses of URs to register themselves. Then based on a given encapsulated ClassID, it can dynamically construct URs of any registered type. Secondly, it can decode a stream containing externalized (i.e. ASCII represented) URs. This class can read a stream containing a UR and identify the UR specified in the stream or the UR Provider referred to by the UR in the stream.

Attributes:

`myStream`

Privilege: Protection Not Identified

Data Type: NOT IDENTIFIED

No Inheritance

This attribute is a pointer to the current input stream associated with this object. This stream contains exported URs that this object helps to import.

Operations:

`DeleteUR (const EcUrUR*)`

Privilege: Protection Not Identified

No Inheritance

This static function is provided as an aid to callers who used `MakeUR` to create a "const EcUrUR*". While they are responsible for deleting it, they can't because it is const. This routine can delete it.

`EcUrURMaker (void)`

Privilege: Protection Not Identified

No Inheritance

This method is the default constructor for the object. The stream still must be set after this constructor.

`EcUrURMaker (const EcUrURMaker&)`

Privilege: Protection Not Identified

No Inheritance

`GetURID (istream&)`

Privilege: Protection Not Identified

No Inheritance

This method returns a reference to the Class ID of the exported UR currently at the beginning of the input stream. The returned class ID will be invalid if the stream does not contain a valid UR.

`GetURProviderID (istream&)`

Privilege: Protection Not Identified

No Inheritance

This method returns a reference to the Class ID of the UR Provider object referred to by the exported UR currently at the beginning of the input stream. An invalid class ID will be returned if the stream does not contain a valid UR.

MakeUR (istream&)

Privilege: Protection Not Identified

No Inheritance

Makes a UR and Internalize it. The user is responsible for deleting the return value through ::DeletedUR. This function can return NULL if the UR defined in the stream is not registered.

Register (const EcUtClassID&, EcUrURProvider*(*func)(),
EcTBoolean replaceOK=FALSE)

Privilege: Protection Not Identified

No Inheritance

~EcUrURMaker (void)

Privilege: Protection Not Identified

No Inheritance

This method is the destructor for the object. It will clean up the state to allow proper deallocation. The state of the internal stream will not be affected by this operation.

5.16.2.4 Class EcUrURProvider

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: EcUrUR(Public)

Class: EcUrURProviderMaker(Public)

Description:

This class is the abstract base class for all things referred to by Universal Reference (UR)s. Its primary responsibility is to provide URs to clients, thus the name "UR Provider". The primary operations of interest are "ProvideUR" and "Reconstitute".

Attributes:

Operations:

CreateUR (void)

Privilege: Protection Not Identified

No Inheritance

This method will create and return an new UR that can refer to this object. This method shall be overridden by all concrete derived classes of this class.

DeleteUR (const EcUrUR*)

Privilege: Protection Not Identified

No Inheritance

This function is static. Since we return a const UR, from ProvideUR, the client can't delete it. This method is responsible for deleting it.

EcUrURProvider (void)

Privilege: Protection Not Identified

No Inheritance

This method is the constructor for the class.

GetMyClassID (void)

Privilege: Protection Not Identified

No Inheritance

ProvideClassUR (EcUrUR&)

Privilege: Protection Not Identified

No Inheritance

Provide primary key data for the class state and place it in the UR. Then, call this method for each of your associated objects.

ProvideUR (void)

Privilege: Protection Not Identified

No Inheritance

This method will provide a Universal Reference to the caller that represents the current logical entity. The return value is allocated on the heap and should be deallocated with the "DeleteUR" method. The return value is a "Memento" (standard design pattern) that can be used to logically bring this object back. The "Reconstitute" method can be called to bring an object back to this state. Note that state is application specific. Derived classes should decide on policies. Possibilities include bring back that exact object the UR came from or reconstituting to the latest version. This method is a "template method" (standard design pattern) that calls "ProvideClassUR".

Reconstitute (const EcUrUR&)

Privilege: Protection Not Identified

No Inheritance

Public method to make ourselves the object that is logically referred to by the UR.

ReconstituteClassData (const EcUrUR&)

Privilege: Protection Not Identified

No Inheritance

Reconstitute class data self based on the UR. Then, call this for associated objects.

~EcUrURProvider (void)

Privilege: Protection Not Identified

No Inheritance

This method is the destructor for the class.

5.16.2.5 Class EcUrURProviderMaker

Synopsis:

No Parent Class

Is Not A Distributed Object

Is Associated With:

Class: EcUrURProvider(Public)

Description:

This class is an object factory responsible for the registration and dynamic creation of object subclasses from "URProvider". Objects are indexed by the encapsulated type "ClassID".

Attributes:

Operations:

EcUrURProviderMaker (void)

Privilege: Protection Not Identified

No Inheritance

This method is the constructor for this object.

MakeURProvider (const EcUR&)

Privilege: Protection Not Identified

No Inheritance

Make a URProvider that matches the UR argument and then Reconstitute it.

Register (const EcClassID&, EcURProvider* (*func)(),
EcTBoolean replaceOK=FALSE)

Privilege: Protection Not Identified

No Inheritance

Register a creation function for a UR Provider derived object.

~EcUrURProviderMaker (void)

Privilege: Protection Not Identified

No Inheritance

This method is the destructor for this method.